Google

continue data protection primary volume secor Search Advanced Search Preferences

Web Results 1 - 10 of about 201,000 for continue data protection primary volume secondary volume. (0.23 seconds)

StorageTek Remote **Volume** Mirroring - 2e2 **Data** Management
Data replication between the **primary volume** and the **secondary volume** is managed by the ... Server applications on the **primary volume continue** during backup ...
www.2e2.com/**data**-management/ds/products/
storagetek/remote**volume**mirroring/mxxrelatedproducts - 25k - Cached - Similar pages

[PDF] HP StorageWorks XP Disk Array and Mainframe white paper
File Format: PDF/Adobe Acrobat - View as HTML
This paper provides an overview of how to achieve **data protection** and **data** consistency across .... **Secondary** site. **Primary** subsystem 1. **Primary data**. **volume** ...
h71028.www7.hp.com/ERC/downloads/4AA1-4114ENW.pdf - Similar pages

[PDF] Using SnapMirror with SnapDrive for UNIX
File Format: PDF/Adobe Acrobat - View as HTML
The **secondary volume** should be of a size equal to or greater than the **primary** system. **volume**. -. The versions of **Data** ONTAP used between the two systems ...
www.netapp.com/library/tr/3611.pdf - Similar pages

[PDF] Snapshot Point-in-time Images for **Data Protection**
File Format: PDF/Adobe Acrobat - View as HTML
writes and reads **continue** on. underlying **data volume** ... **primary volume**. In addition, SBE's. Snapshot module is highly scalable and ...
www.sbei.com/support_files/**data**sheets/SBE_IPSAN_Snapshot_**Data**sheet_v1.2.pdf - Similar pages

Remote copy **secondary data** copy validation-audit function - US ...
Hence still further **protection** is required for recovering **data** if a disaster occurs .... When a **secondary volume** is out of sync with a **primary volume**, ...
www.patentstorm.us/patents/5592618-description.html - 73k - Cached - Similar pages

SGI TPL (Linux: End-User/TPSconcepts - Chapter 3. Remote **Volume** ...
Whenever the **data** on the **primary volume** and the **secondary volume** becomes ... The **primary** host can **continue** to write to the **primary volume** but remote writes ...
techpubs.sgi.com/.../cgi-bin/getdoc.cgi?coll=linux&
db=bks&fname=/SGI_EndUser/TPSconcepts/ch03.html - 232k - Cached - Similar pages

Fault Tolerance
Mirrored volumes provide an identical copy for a selected **volume**. All **data** that is written to the **primary volume** is also written to a **secondary volume** or ...
www.microsoft.com/technet/prodtechnol/
windows2000serv/reskit/deploy/dgbj_sto_njom.mspx - 10k - Cached - Similar pages

Incentra Solutions
Simply define to PowerPath that the **primary volume** (source) is the local disk and that the ... State of Technology: Getting Creative with **Data Protection** ...
www.star-solutions.com/interior.php/sid/37/aid/159/pid/210/nid/51 - 96k - Cached - Similar pages

Remote copy **secondary data** copy validation-audit function - Patent ...
When a **secondary volume** is out of sync with a **primary volume**, the **secondary** ..... The **secondary** storage system will **continue** to update **data** copies for the ...
www.freepatentsonline.com/5592618.html - 87k - Cached - Similar pages

SNIA - Dictionary R
Within a redundancy group, a single type of **data protection** is employed. .... A replication set consists of a **primary volume** and a **secondary volume** that are ...
www.snia.org/education/dictionary/r/ - 60k - Cached - Similar pages

1 2 3 4 5 6 7 8 9 10 **Next**

---

continue data protection primary vol Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve | Try Google Experimental

---

Google

continue data protection primary volume secor Search Advanced Search Preferences

Web Results 11 - 20 of about 201,000 for continue data protection primary volume secondary volume. (0.20 seconds)

Translator: MagicISO, Inc. ;---- Date: 10-04-2006 ;---- EMail ...
Caption = 'Correct the serial number of **Primary Volume** Descriptor' ..... 264 = "Failed to write **data** to file, Please make sure the target driver is not full ...
www.magiciso.com/language/language.ini - 36k - Cached - Similar pages

Blackwell Synergy - JDDG, **Volume** 5 Issue 9 Page 756-760, September ...
Skin **protection** and skin care are the main components of prevention of these diseases. ... Despite knowledge of the special role of **primary** and **secondary** ...
www.blackwell-synergy.com/doi/abs/10.1111/j.1610-0387.2007.06434.x - Similar pages

NEA News - 2004 **Volume** 22, Number 1: Uranium production and demand ...
Primary production, therefore, only provided about 54% of world reactor requirements at the end of 2002. The remaining demand was met using **secondary** ...
www.nea.fr/html/pub/newsletter/2004/22-1-uranium.html - 17k - Cached - Similar pages

The Hungarian Quarterly, **VOLUME** XLI * No. 160 * Winter 2000
Before citing the **data** I must stress that our survey concerned general (**primary**) schools, we obtained no **data** on **secondary** school-leaving-certificate ...
www.hungarianquarterly.com/no160/091.html - 24k - Cached - Similar pages

[PDF] RVM_AAG (Page 1)
File Format: PDF/Adobe Acrobat - View as HTML
data protection efforts at their **primary** site while adding seamless, synchronous mirroring at. a second location. BlueArc Remote **Volume** Mirroring (RVM) ...
www.bluearc.com/html/library/downloads/rvm_ds.pdf - Similar pages

Journal of Vascular Surgery : **Volume** regression of abdominal ...
A **volume** decrease of 10% or greater at 6 months and **continuing** regression over time is associated with successful endovascular repair and **protection** from ...
linkinghub.elsevier.com/retrieve/pii/S0741521403009248 - Similar pages

[PDF] Splash NW-35_252 2
File Format: PDF/Adobe Acrobat - View as HTML
Updates **continue** on the **primary volume** while. the **secondary volume** is being backed up. ... corporations is Hitachi **Data** Systems' **primary** strength. ...
www.hitachi**data**systems.com/download.html?url=/pdf/wp136_backup_restore.pdf&region=global&id=123... - Similar pages

[PDF] Continuous **data protection**: addressing timely **data** recovery and ...
File Format: PDF/Adobe Acrobat - View as HTML
be either block (**volume**) or file based. Block-based solutions operate at the block. level of logical devices. As **data** blocks are written to **primary** storage, ...
material.talentum.com/tietoviikko_**data**/tivi_infra_pdfs/Continuous_**data_protection**_2005.pdf - Similar pages

InfoStor - Disk-to-disk-to-tape vs. replication, part 1
Local mirrors are physical full **volume** copies of the **data** within a .... A **primary** difference between **secondary** disk storage and disk libraries is the lack ...
www.infostor.com/articles/article_display.cfm?article_id=214702 - 74k - Cached - Similar pages

[PDF] Examining Hitachi Copy-on-Write Snapshot Software Capabilities
File Format: PDF/Adobe Acrobat - View as HTML

Restore Operation. A restore operation writes (returns) **secondary volume** (Snapshot Image) **data** that has been retained as. a backup into a **primary volume**; ...
www.virtual.com/whitepapers/HDS_Examining_Copy-on-Write.pdf - Similar pages

Previous 1 2 3 4 5 6 7 8 9 10 11 **Next**

---

continue data protection primary vol Search

Search within results | Language Tools | Search Tips | Try Google Experimental

---

Web Results **1** - **10** of about **58,800** for **continue data protection primary volume secondary volume APIT any point in time**.

Method and system for **data** recovery in a continuous **data ...**
The **data protection** system 106 manages a **secondary data volume** 108 . ... at a **time** to recover the state of the **primary volume** at **any** previous **point in time**. ...
www.freepatentsonline.com/7325159.html - 90k - Cached - Similar pages

Remote management commands in a mass storage system - Patent 7302604
A PiT copy is generated either at the **primary** or at the **secondary** .... An example of a **data** storage management command is a **point-in-time** (PiT) copy command ...
www.freepatentsonline.com/7302604.html - 51k - Cached - Similar pages
More results from www.freepatentsonline.com »

Remote management commands in a mass storage system - US Patent ...
A PiT copy is generated either at the **primary** or at the **secondary** facility, ... directly accesses a **primary volume**, and **data** written to a **primary volume** is ...
www.patentstorm.us/patents/7302604-description.html - 41k - Cached - Similar pages

[PDF] The Hitachi NanoCopy Advantage — An Industry First for **Point-in ...**
File Format: PDF/Adobe Acrobat - View as HTML
For the first **time** in the industry, **a PiT** copy of **any** amount of **data** can be .... **Primary** Host. System **Data**. Mover Host. **Secondary**. Logical **Volume**. **Primary** ...
www.hitachi**data**systems.com/download.html?url=/pdf/wp134_nanocopy.pdf&region=global&id=511&typ... - Similar pages

[PDF] Disaster Recovery Issues and Solutions
File Format: PDF/Adobe Acrobat - View as HTML
Two discrete **points in time** define **any** catastrophic disaster: when the disaster ..... **Data** + T. ime Stamp. **Secondary**. Logical. **Volume**. **Data + Time** Stamp ...
infoweek.ch/dossiers/files/Disaster_recovery_Hitachi.pdf - Similar pages

StorageTek Snapshot copy - 2e2 **Data** Management
Data Protection – The **time** and cost to backup **data** is a major consideration, ... Server applications utilising the **primary volume continue** during backup ...
www.2e2.com/**data**-management/ds/products/storagetek/snapshotcopy/mxxrelatedproducts - 24k - Cached - Similar pages

(WO/2007/002397) SYSTEM AND METHOD FOR HIGH PERFORMANCE ENTERPRISE ...
A "Double **Protection**" feature is provided whereby **point-in-time** images in the ..... A **primary volume** may be mirrored onto a **secondary volume** in accordance ...
www.wipo.int/pctdb/en/wo.jsp?wo=2007002397&IA=WO2007002397&DISPLAY=DESC - 97k - Cached - Similar pages

[PDF] Ensuring **Data** Integrity with Asynchronous Replication
File Format: PDF/Adobe Acrobat - View as HTML
To mitigate this risk, **a PiT** copy from the **secondary volume** should be snapped .... But from a **data-protection point** of view, the most important goal is to ...
www.hds.co.uk/assets/pdf/wp_200_**data**_integrity_asynch_rep.pdf - Similar pages

[PDF] Disaster Recovery Issues and Solutions
File Format: PDF/Adobe Acrobat - View as HTML
Two discrete **points in time** define **any** catastrophic disaster: when the disaster ..... **Primary** Host. **Primary**. Logical. **Volume**. **Data + Time** Stamp. **Secondary** ...
www.hds.com/pdf/wp_117_02_disaster_recovery.pdf - Similar pages

[PDF] Survey to take on international focus as **APIT**
File Format: PDF/Adobe Acrobat - View as HTML
were to investigate **primary** and. **secondary** sources of information on the. history, origin and meaning of NSW. placenames, evaluate the **volume** and ...
www.anps.org.au/documents/Dec_2001.PDF - Similar pages

1 2 3 4 5 6 7 8 9 10 **Next**

---

continue data protection primary voli Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve | Try Google Experimental

---

Web Results **11** - **20** of about **58,800** for **continue data protection primary volume secondary volume APIT any point in time**

SNS EUROPE -
Volume manager mirroring is generally used for local high availability rather than **data** replication for **point-in-time** copies. It is usually implemented in ...
www.snseurope.com/snslink/news/articles-full.php?newsid=4675&department=Software - 75k - Cached - Similar pages

[PDF] Usage Guide for Sun StorEdge Instant Image Software with Oracle 8
File Format: PDF/Adobe Acrobat - View as HTML
Using Sun StorEdge Instant Image Software to Create a **Point-in-Time Data**. Copy 43. Using the Shadow **Volume** Resulting from **a PIT** Copy Operation 45 ...
www.oracle.com/technology/deploy/availability/pdf/sun_snap_usage.pdf - Similar pages

[PDF] PIT AND BELOW-GRADE TANK GUIDELINES
File Format: PDF/Adobe Acrobat - View as HTML
ground level, or if **a pit volume** is more than 10 acre-feet, ... At **any point** of discharge into the pit, the discharge shall be directed away ...
www.emnrd.state.nm.us/MAIN/documents/PITandBelowGradeTankGuidelines.pdf - Similar pages

[PDF] P R O D U C T P R O F I L E DataCore SANsymphony 6.0 – The Perfect ...
File Format: PDF/Adobe Acrobat - View as HTML
providing an **Any-Point-In-Time** (**APIT**) copy. of enterprise **data**. ... Further abilities to structure **volume**. characteristics such as **protection** and ...
www.datacore.com/downloads/SANSymphony%206%200%20Product%20Profile%20-%20March%202007%20-%20Final.pdf - Similar pages

Terms Beginning With "S"
Specific Yield: The amount of water a unit **volume** of saturated permeable rock ... Start of a Response Action: The **point in time** when there is a guarantee or ...
www.epa.gov/OCEPAterms/sterms.html - 47k - Cached - Similar pages

[PDF] HP StorageWorks XP Disk Arrays and Mainframe white paper: **Data** ...
File Format: PDF/Adobe Acrobat - View as HTML
mirrored copy contains the **primary** and **secondary data**, there is although no disaster security ..... Real-**time** local mirroring and/or snapshots at **volume** and ...
h71028.www7.hp.com/ERC/downloads/4AA0-3228ENW.pdf - Similar pages

[PDF] Hitachi Software Guide-front
File Format: PDF/Adobe Acrobat - View as HTML
secondary logical **volume** (as might happen in a rolling disaster! ..... For the first **time** in the industry, **a PiT** copy of **any**. amount of **data** can be created, ...
www.iarchive.com/_library/whitepapers/_articles/ssg.pdf - Similar pages

[PDF] 7. CASE STUDIES
File Format: PDF/Adobe Acrobat - View as HTML
chamber for the **secondary** outlet. The water level will **continue** to rise until the required detention **volume** has been. achieved. **Any** overflows caused by ...
www.uprct.nsw.gov.au/osd/2005_4th%20edition/Chapter%207.pdf - Similar pages

[PDF] A V CA – Afr ican V entur e C **apit** al Associa tion Dir ect or y 2004

File Format: PDF/Adobe Acrobat - View as HTML
It can be dissolved at **any time** in accor- ..... Cumulative quarterly trading **volume** during last quarter (no. shares). 25 000 000 ...

www.avcanet.com/publications/AVCA%20Electronic%20Directory.pdf - Similar pages

Annals of Botany - Fulltext: **Volume** 98(3) September 2006 p 483-494 ...
Such vascular organization is thought to act as a **protection** against the ... Ye and the beads moved freely from the stem into **primary** and **secondary** veins of ...
pt.wkhealth.com/pt/re/abot/fulltext.00008707-200609000-00003.htm - Similar pages

Previous 1 2 3 4 5 6 7 8 9 10 11 **Next**

Web Results **21** - **30** of about **58,800** for **continue data protection primary volume secondary volume APIT any point in time**

[PDF] Oracle Database 10g Automatic Storage Management Best Practices ...
File Format: PDF/Adobe Acrobat - View as HTML
Consistency Groups and At-**Time** Split Options for **Point-in-Time** Copies . ..... **Primary**. **Data**. **Volume**. Storage Navigator PC. Universal Storage Platform ...
www.oracle.com/.../**data**base/asm/pdf/HDS%20PIT%20backup-recovery%20%20BP%20with%20ASM%2007-2007_0.pdf - Similar pages

Zirconia-alumina nanolaminate for perforated pitting corrosion ...
The **protection** afforded by 250 nm thick films with two nanolaminate architectures, ... the reduction in layer thickness and the increase in **volume** fraction ...
link.aip.org/link/?JVTAD6/22/272/1 - Similar pages

Blackwell Synergy - Botan J Linn Soc, **Volume** 150 Issue 1 Page 115 ...
Secondary and **primary** walls were again replaced by mineral. ..... Thus, smaller numbers and diameters of tracheids will also reduce **volume** and rate of water ...
www.blackwell-synergy.com/doi/abs/10.1111/j.1095-8339.2006.00450.x - Similar pages

Final Environmental Impact Statement for the **Continued** Operation ...
Aircraft accidents are a concern at Pantex Plant because of the **volume** of local air ... as separate magazines) can be open at **any** one **time** (DOE 1992b:8-21). ...
www.globalsecurity.org/wmd/library/report/enviro/eis-0225/eis0225_415.html - 46k - Cached - Similar pages

[PDF] Section 4c - Sanitation.p65
File Format: PDF/Adobe Acrobat
will not function; if there is a high **volume** of water supplied, ... The approximate **time** taken to fill **a pit** can be estimated using the **data** in. Table S3. ...
www.lboro.ac.uk/wedc/publications/sftup/sftup-4c-ref.pdf - Similar pages

Terms Beginning With S (Begriffe S)
Signal: The **volume** or product-level change produced by a leak in a tank. .... Start of a Response Action: The **point in time** when there is a guarantee or ...
www.waterquality.de/hydrobio.hw/STERMS.HTM - 39k - Cached - Similar pages

[PDF] Proposed Acceptability for **Continuing** Registration PACR2005-08
File Format: PDF/Adobe Acrobat - View as HTML
that the use of strychnine does result in **primary** and **secondary** poisoning of non-target. species. ..... **Volume** 1 of 2. Office of Health and Environmental ...
www.pmra-arla.gc.ca/english/pdf/pacr/pacr2005-08-e.pdf - Similar pages

Current Opinion in Endocrinology, Diabetes and Obesity - Fulltext ...
Current Opinion in Endocrinology & Diabetes:**Volume** 11(6)December 2004pp 330- ..... osteoclasts are recruited to the surface of the bone and create **a pit** in ...
www.co-endocrinology.com/pt/re/coendo/fulltext.00060793-200412000-00003.htm;jsessionid=GkKZzyf1H4YJ7zrKGX... - Similar pages

[PDF] DOE-HDBK-1015/1-92; DOE Fundamentals Handbook Chemistry **Volume** 1 of 2
File Format: PDF/Adobe Acrobat - View as HTML
U F in the process gas at **any point** in the cascade. These calculations can be made by ..... of metals used in the various **primary** and **secondary** systems. ...
hss.energy.gov/NuclearSafety/techstds/standard/hdbk1015/h1015v1.pdf - Similar pages

[PDF] Global Mirror whitepaper

File Format: PDF/Adobe Acrobat - View as HTML
Figure 27 **Volume** layout with two drive sizes on **secondary** disk subsystem. ..... pause the session in preparation for the next **time a PiT** copy is desired. ...
www-03.ibm.com/systems/storage/solutions/business_continuity/pdf/globalmirrorwp.pdf - Similar pages

Previous 1 2 3 4 5 6 7 8 9 10 11 12 **Next**

continue data protection primary vol Search

Search within results | Language Tools | Search Tips | Try Google Experimental

Google

continue data protection primary volume secor Search Advanced Search Preferences

Web

Your search - **continue data protection primary volume secondary volume APIT any point in time block-ordered** - did not match any documents.

Suggestions:

- Make sure all words are spelled correctly.
- Try different keywords.
- Try more general keywords.
- Try fewer keywords.

Web Results **1 - 10** of about **39,600** for **continue data protection primary volume secondary volume APIT any point in time b**

Method and system for **data** recovery in a continuous **data ...**
The **data protection** system 106 manages a **secondary data volume** 108 . ... at a **time** to recover the state of the **primary volume** at **any** previous **point in time**. ...
www.freepatentsonline.com/7325159.html - 90k - Cached - Similar pages

Remote management commands in a mass storage system - Patent 7302604
A PiT copy is generated either at the **primary** or at the **secondary** .... An example of a **data** storage management command is a **point-in-time** (PiT) copy command ...
www.freepatentsonline.com/7302604.html - 51k - Cached - Similar pages
More results from www.freepatentsonline.com »

Remote management commands in a mass storage system - US Patent ...
A PiT copy is generated either at the **primary** or at the **secondary** facility, ... directly accesses a **primary volume**, and **data** written to a **primary volume** is ...
www.patentstorm.us/patents/7302604-description.html - 41k - Cached - Similar pages

(WO/2007/002397) SYSTEM AND METHOD FOR HIGH PERFORMANCE ENTERPRISE ...
A "Double **Protection**" feature is provided whereby **point-in-time** images in the ..... A **primary volume** may be mirrored onto a **secondary volume** in accordance ...
www.wipo.int/pctdb/en/wo.jsp?wo=2007002397&IA=WO2007002397&DISPLAY=DESC - 97k - Cached - Similar pages

SNS EUROPE -
Volume manager mirroring is generally used for local high availability rather than **data** replication for **point-in-time** copies. It is usually implemented in ...
www.snseurope.com/snslink/news/articles-full.php?newsid=4675&department=Software - 75k - Cached - Similar pages

[PDF] Usage Guide for Sun StorEdge Instant Image Software with Oracle 8
File Format: PDF/Adobe Acrobat - View as HTML
Using Sun StorEdge Instant Image Software to Create a **Point-in-Time Data**. Copy 43. Using the Shadow **Volume** Resulting from **a PIT** Copy Operation 45 ...
www.oracle.com/technology/deploy/availability/pdf/sun_snap_usage.pdf - Similar pages

[PDF] Hitachi Software Guide-front
File Format: PDF/Adobe Acrobat - View as HTML
set **any time** for **point-in-time** copies to be taken, without **any** outage to the ..... **order** to have consistent, **volume**-level backup of **data**, all applications ...
www.iarchive.com/_library/whitepapers/_articles/ssg.pdf - Similar pages

Annals of Botany - Fulltext: **Volume** 98(3) September 2006 p 483-494 ...
Such vascular organization is thought to act as a **protection** against the ... Ye and the beads moved freely from the stem into **primary** and **secondary** veins of ...
pt.wkhealth.com/pt/re/abot/fulltext.00008707-200609000-00003.htm - Similar pages

Zirconia-alumina nanolaminate for perforated pitting corrosion ...
The **protection** afforded by 250 nm thick films with two nanolaminate architectures, ... the reduction in layer thickness and the increase in **volume** fraction ...
link.aip.org/link/?JVTAD6/22/272/1 - Similar pages

[PDF] Section 4c - Sanitation.p65
File Format: PDF/Adobe Acrobat
will not function; if there is a high **volume** of water supplied, ... The approximate **time**

taken to fill **a pit** can be estimated using the **data** in. Table S3. ...
www.lboro.ac.uk/wedc/publications/sftup/sftup-4c-ref.pdf - Similar pages

1 2 3 4 5 6 7 8 9 10 **Next**

continue data protection primary vol Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve | Try Google Experimental

Web Results **11** - **20** of about **39,600** for **continue data protection primary volume secondary volume APIT any point in time**

Blackwell Synergy - Biol J Linn Soc, **Volume** 76 Issue 2 Page 165 ...
Order, Nature of **primary data** from which trees were constructed, **Data** sources .... Biological Journal of the Linnean Society, **Volume** 85, Issue 3, ...
www.blackwell-synergy.com/doi/abs/10.1046/j.1095-8312.2002.00055.x - Similar pages

[PDF] DOE-HDBK-1015/1-92; DOE Fundamentals Handbook Chemistry **Volume** 1 of 2
File Format: PDF/Adobe Acrobat - View as HTML
U F in the process gas at **any point** in the cascade. These calculations can be made by ..... of metals used in the various **primary** and **secondary** systems. ...
hss.energy.gov/NuclearSafety/techstds/standard/hdbk1015/h1015v1.pdf - Similar pages

IEEE Std 980-1994 (R2001, Revision of IEEE Std 980-1987) IEEE ...
primary oil containment, retention pit, **secondary** oil containment, ... 3.6 oil discharge: **Any** leak or spillage of oil, regardless of **volume** and including ...
ieeexplore.ieee.org/iel1/3265/9816/00467459.pdf - Similar pages

[PDF] H A D R — O 10 & 9 A F C
File Format: PDF/Adobe Acrobat
volume in **order** to configure them for a different synchronization level than the ..... corresponding abilities to restore **data** to **Any Point in Time** (**APIT**), ...
www.natcapoug.org/presntn_downloads/HADR-OraVsCmptn_JeffB+DrRanP_v20060328a.050426x-RstrctCpy.pdf - Similar pages

[PDF] The Class V Underground Injection Control Study **Volume** 5 Large ...
File Format: PDF/Adobe Acrobat - View as HTML
If at **any point** during construction the soil is damaged by smearing, compaction, ... meet in **order** to provide long-term **protection** of ground water. ...
www.epa.gov/ogwdw/uic/class5/pdf/study_uic-class5_classvstudy_**volume**05-largecapacitysepticsystems.pdf - Similar pages

[PDF] Common Ground Journal **Volume** 5 Number 1: Generations
File Format: PDF/Adobe Acrobat - View as HTML
games in which game rules are tested to the breaking **point** in **order** to learn ..... past century idea of a **secondary** nature. I would argue for the **data** that ...
www.commongroundjournal.org/volnum/v05n01.pdf - Similar pages

Glossary - Itron
Any data that is gathered and stored by the meter can be communicated via the ..... The "dials" on the front of a meter which indicates the **volume** of gas ...
www.itron.com/pages/resources_glossary.asp - 176k - Cached - Similar pages

[PDF] DMRB **VOLUME** 6 SECTION 3 PART 3 - TA 57/87 - ROADSIDE FEATURES
File Format: PDF/Adobe Acrobat - View as HTML
PEDESTRIAN GUARDRAILING - **PROTECTION** BY SAFETY FENCING. Layouts for existing sites where space is restricted. FIG.6. Crossing. **Volume** 6 Section 3. Chapter 4 ...
www.standardsforhighways.co.uk/dmrb/vol6/section3/ta5787.pdf - Similar pages

[PDF] Using the SVC for Business Continuity
File Format: PDF/Adobe Acrobat
secondary by writing **data** in the same **order** it happened on **primary**. ..... **Volume** Shadows copy Service. (VSS) ensures the **data** consistency of **a PiT** ...
www.redbooks.ibm.com/redbooks/pdfs/sg247371.pdf - Similar pages

[PDF] Solomon Islands Smallholder Agriculture Study: **Volume** 2
File Format: PDF/Adobe Acrobat - View as HTML
Volume 2, Subsistence Production, Livestock and Social. Analysis, covers livestock issues
(Ben ..... have the necessary free **time** to do so. **Any** income- ...
www.ausaid.gov.au/publications/pdf/solomon_study_vol2.pdf - Similar pages

Previous 1 2 3 4 5 6 7 8 9 10 11 **Next**

Web Results **1** - **10** of about **4,370** for **block ordered continue data protection primary volume secondary volume APIT any** |

Method and system for **data** recovery in a continuous **data** ...
The **data protection** system 106 manages a **secondary data volume** 108 . ... at a **time** to recover the state of the **primary volume** at **any** previous **point in time**. ...
www.freepatentsonline.com/7325159.html - 90k - Cached - Similar pages

Remote management commands in a mass storage system - Patent 7302604
A PiT copy is generated either at the **primary** or at the **secondary** .... An example of a **data** storage management command is a **point-in-time** (PiT) copy command ...
www.freepatentsonline.com/7302604.html - 51k - Cached - Similar pages
More results from www.freepatentsonline.com »

Remote management commands in a mass storage system - US Patent ...
A PiT copy is generated either at the **primary** or at the **secondary** facility, ... directly accesses a **primary volume**, and **data** written to a **primary volume** is ...
www.patentstorm.us/patents/7302604-description.html - 41k - Cached - Similar pages

(WO/2007/002397) SYSTEM AND METHOD FOR HIGH PERFORMANCE ENTERPRISE ...
A "Double **Protection**" feature is provided whereby **point-in-time** images in the ..... A **primary volume** may be mirrored onto a **secondary volume** in accordance ...
www.wipo.int/pctdb/en/wo.jsp?wo=2007002397&IA=WO2007002397&DISPLAY=DESC - 97k - Cached - Similar pages

SNS EUROPE -
Volume manager mirroring is generally used for local high availability rather than **data** replication for **point-in-time** copies. It is usually implemented in ...
www.snseurope.com/snslink/news/articles-full.php?newsid=4675&department=Software - 75k - Cached - Similar pages

[PDF] Usage Guide for Sun StorEdge Instant Image Software with Oracle 8
File Format: PDF/Adobe Acrobat - View as HTML
Using Sun StorEdge Instant Image Software to Create a **Point-in-Time Data**. Copy 43. Using the Shadow **Volume** Resulting from **a PIT** Copy Operation 45 ...
www.oracle.com/technology/deploy/availability/pdf/sun_snap_usage.pdf - Similar pages

[PDF] Hitachi Software Guide-front
File Format: PDF/Adobe Acrobat - View as HTML
set **any time** for **point-in-time** copies to be taken, without **any** outage to the ..... **order** to have consistent, **volume**-level backup of **data**, all applications ...
www.iarchive.com/_library/whitepapers/_articles/ssg.pdf - Similar pages

Annals of Botany - Fulltext: **Volume** 98(3) September 2006 p 483-494 ...
Such vascular organization is thought to act as a **protection** against the ... Ye and the beads moved freely from the stem into **primary** and **secondary** veins of ...
pt.wkhealth.com/pt/re/abot/fulltext.00008707-200609000-00003.htm - Similar pages

Zirconia-alumina nanolaminate for perforated pitting corrosion ...
The **protection** afforded by 250 nm thick films with two nanolaminate architectures, ... the reduction in layer thickness and the increase in **volume** fraction ...
link.aip.org/link/?JVTAD6/22/272/1 - Similar pages

[PDF] Section 4c - Sanitation.p65
File Format: PDF/Adobe Acrobat
will not function; if there is a high **volume** of water supplied, ... The approximate **time**

taken to fill **a pit** can be estimated using the **data** in. Table S3. ...
www.lboro.ac.uk/wedc/publications/sftup/sftup-4c-ref.pdf - Similar pages

1 2 3 4 5 6 7 8 9 10 **Next**

(WO/2007/002397) SYSTEM AND METHOD FOR HIGH PERFORMANCE ENTERPRISE DATA PROTECTION

Note: OCR Text

TITLE: System And Method for High Performance Enterprise Data Protection

CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of the filing date of U.S. Provisional Application No. 60/693,715, filed on June 24, 2005, which is incorporated herein by reference. This application also incorporates by reference the entire disclosure of our commonly invented and commonly assigned application entitled "System And

Method for Virtualizing Backup Images", Application No. , which is being filed on the same date as this application.

BACKGROUND OF THE INVENTION

Field of the Invention The present invention is in the field of information technology, and more particularly relates to high performance, enterprise-level backup and disaster recovery systems.

Description of Related Art

Recent events have proved that the need to recover quickly from disasters (both man-made and natural) is critical. Enterprise-level backup and disaster recovery systems are directed at this need. Under the current state of the art, the typical end product of a backup operation is a backup volume that must go through a relatively lengthy "restore" process before it can be used in production.

There do exist some "short downtime" backup and recovery solutions, but they generally require expensive server clustering and/or replication capabilities.

The state of the art with respect to the present application is documented in the publications of the Storage Networking Industry Association ("SNIA"), which are accessible online at www.snia.orK. See in particular "Examination of Disk Based

Data Protection Technologies" by Michael Rowan, of Revivio Corporation; "Identifying and Eliminating Backup System Bottlenecks" by Jacob Farmer of

Cambridge Computer Corporation; "Technologies to Address Contemporary Data Protection" by Michael Fishman of EMC Corporation; and "Next Generation Business Continuity" by Andrea Chiaffitelli of AT&T Corp. (each of which references is incorporated by reference). As will be appreciated form a review of the references cited above, the current state of the art does not provide a method short of large-scale server clustering and/or replication for making recent point-in-time snapshots of a system available for use on an immediate basis in the event of a system failure or disaster.

It would be desirable, therefore, to have a system implemented with simple hardware that provides the capability so that an organization at any given time could have a recent set of self-consistent images of its production servers available that, in the event of a system failure or disaster, could be brought online and into active production on a more-or-less instantaneous basis.

SUMMARY OF THE INVENTION An embodiment of the present invention is being made available as part of

Backup Express® (BEX), a software product of Syncsort Incorporated, the assignee of the present application. Among other capabilities, the present invention, as implemented in Backup Express, provides a service called "Fast Application Recovery" (FAR), which makes possible near instant recovery from failure using simple hardware well within the IT budgets of most businesses.

It is an object of the present invention to provide a high performance, enterprise-level data protection system and method providing efficient block-level incremental snapshots of primary storage devices, and instant availability of such snapshots in immediately mountable form that can be directly used in place of the primary storage device.

Among other objects of the invention are the following:

• providing an enterprise repository for such snapshots adapted to facilitate the methods described herein on a variety of storage platforms. • providing the ability create a replacement physical primary facility in real time while working with another storage unit as the primary.

• providing the ability to eliminate redundancy in multiple backups and/or in a single file system by means of block level comparisons.

In one embodiment, the instant availability aspect of the invention is provided by: a) providing a base-level snapshot, stored on a secondary system of the source ("primary") file system; b) providing a block-level incremental snapshots of the primary system, stored on the secondary system, representing only the blocks that have changed since the prior snapshot; and c) constructing a logical disk image from at least one of said incremental snapshot images that can be used directly as a mounted storage unit (the incremental snapshot in step b having been constructed in a manner that facilitates the immediate performance of this step on demand).

The snapshotting and instant availability features of the invention are used in connection with storage hardware components to provide an "Enterprise Image Destination" (EID) for backup images created in accordance with the present invention. The EID software is further distinguished in being operable with storage hardware from a wide variety of vendors, including inexpensive ATA storage hardware. A "Double Protection" feature is provided whereby point-in-time images in the EID may themselves be backed up to selected media or replicated in other EIDs.

The invention also provides a feature, referred to as "lazy mirroring," whereby a replacement physical primary facility may be created while working with a second storage unit as the primary source file system. The second storage unit in accordance with this feature could be a secondary logical volume previously brought online pursuant to the "instant availability" feature of the invention referenced above, where a replacement physical primary volume is being created at the same time; or it could be (as another example) a surviving unit of a mirrored storage system where another mirror unit is being "resilvered" or replaced at the same time. Other applications of the "lazy mirroring" technique are possible as well. In any such application, the "lazy mirroring" in accordance with the invention is further characterized by being able to proceed without an interruption in processing.

Finally, the invention provides a technique based on block comparisons for greatly speeding up distributed backup operations by eliminating redundant data when multiple systems with partially common contents (e.g., operating system files and

common databases) are being backed up. Where it is determined that a block to be backed up already exists in the backup set, the existing block is used in the directory or catalog of the backup, rather than storing both blocks. A similar technique is employed so as to eliminate redundant blocks in a file system. Other objects and advantages of the invention will be clear from the drawings and the detailed description which follows.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a high level system block diagram showing a typical enterprise deployment of an embodiment of the invention. Fig. 2 is a block diagram showing block-level backup data transfer and file- level restore.

Fig. 3 is a block diagram showing a time line of operations that are part of a block-level incremental backup, followed by an exemplary file-level restore.

Fig. 4(A & B) is a block diagram showing a time line of an example disaster recovery scenario involving incremental block-level backup, instant availability restore, and "lazy mirror" replication.

DETAILED DESCRIPTION

The following is a description of several preferred embodiments of various aspects of the invention, showing details of how systems may be constructed to carry out the invention, and the steps that can be employed to utilize such systems and to practice such methods. These embodiments are illustrative only, and the invention is by no means limited to particular examples shown. For example, certain preferred embodiments are described in relation to an implementation with specific storage hardware and operating systems, but it should be appreciated that the disclosure that follows was intended to enable those skilled in the art readily to apply the teachings set forth to other storage hardware and operating systems. The specific features of any particular embodiment should not be understood as limiting the scope of what may be claimed.

DEFINITIONS The following terms have a defined meaning as used in this application:

APM (Advanced Protection Manager): A name used for a suite of products that implement an embodiment of the present invention.

APM2D (Advanced Protection Manager to Disk): An umbrella term covering presently available secondary devices, and future solutions in a system that provides forever block level incrementals and Instant Availability.

Application: A mass produced (i.e., generally commercially licensed) back- end to a business application (usually a database) that is protected by backup. This is distinct from (and should not be confused with) the "end user application.

Application Instance: A logically separate incarnation of an application coexisting with other instances on a physical machine. An application instance is the target for FAR. Backup Client: Client software that provides block-level incremental backup for high-speed backup with virtually no impact on other operations. Accesses the disk directly, bypassing the file system for extremely fast, efficient image-based backups. Backup Clients are also provided for block-level incremental backup of Exchange 2000/2003 and SQL Server 2000 databases. BAR (Backup After Restore): The first backup after restore is also an incremental and is tied to the original base.

EID (Enterprise Image Destination): Nearline destination and repository for application-aware Forever Image Incrementals.

EOFM: - OEM version of snapshot driver from St. Bernard for Windows. ERF (Eventual Rapid Failback) for Applications: It may be desirable to fallback the application from the target node for FAR back to the original or newly designated home node for the application. This is performed rapidly, seamlessly with minimum application downtime.

ExpressDR: Provides simple, robust one-step bare metal recovery for client nodes from routine daily backups. Can also be used to deploy a complete system image to multiple machines.

Express Image: Utilizes block-level technology for high-performance backup of systems to tape or storage-independent disk. Provides exceptional performance gains for high-volume backups with many small files. FAR (Fast Application Recovery): Fast Application Recovery is the ability to bring an application on-line quickly on a stand-by or original server by attaching to virtual storage created out of backup images on a NAS device.

Filer: a NAS device.

Forever Image Incrementals (also, "Forever incrementals" and "Forever block-level incrementals"): The ability to seed base level back and then schedule incremental, block-level backups forever thereafter.

Instant Availability: Enables rapid mounting of backup data sets as read/write volumes. Provides near-instant recovery of critical applications and data without transferring data. iSCSI: TCP/IP based protocol for storage. Low cost alternative to fiber channel for making remote storage on an IP network accessible to any authenticated initiator node. iSCSI mapping and Unmapping: The process of iSCSI login to the filer makes LUNs on the filer visible as local storage on the restore target node. iSCSI logoff undoes this process and removes these disks.

LAR (Life After Restore): This is a combination of ERP and Backup of the FAR volumes if there is business value in protecting the FAR volumes. LUN Cloning: A feature of NAS filers which allows snapshot backed LUNs to be freed from the backing snapshot and transition to a normal LUN. The LUN can be used by applications while this process completes. The snapshot can then be deleted and the LUN has independent existence.

LUN Creation: A feature of a NAS filer carving virtual storage out of backup images stored in snapshots. These LUNs can then be mounted read-write on the restore target. Reads are satisfied from the snapshot while writes are directed to a separate persistent area. The original backup image does not change.

Online/Background Restore: Automatic background copying of image data from iSCSI drives to a local disk slice, following FAR, while the application remains online. This is done un-obtrusively in the background while the application is up and running. A short synchronization is needed at the end when the application is quiesced or restarted and the iSCSI drive unmapped. At the end of the process all data is local. No penalty is paid in terms of application outage or downtime while data transfer happens. PIT Images: Point in time Images of application volumes, frozen at the time of backup.

Protocol director: Controls and manages the execution of jobs employing a block-level application-consistent protocol.

Secondary Storage: Distinct from primary storage (which is where production data resides) this is the destination for backup as well as the bedrock for LUNs that form virtual machine disks. Only changes require additional storage, thus little secondary storage beyond what is necessary for backup is needed. This storage may be Write Once Read Many (WORM) to support un-alterable content retention to meet legal requirement.

Specialized Backup Software: This creates backup images capturing incremental changes and preserving points in time in the past on secondary storage. Backup software creates application consistent images and additionally captures machine configuration including persistent and volatile state.

Application Manager: Manages all block-level application consistent backup operations from an easy-to-use, browser-based GUI. Supports backup of NAS devices plus Windows, UNIX, and Linux nodes. Also displays SQL and Exchange volumes and databases in the GUI for selectable backup and restore. All backups and other operations are tracked in a single catalog.

Stand-by Node/ Alternate Node/Preventive Setup: A machine with minimal hardware and default application installation which could be the target for FAR for high availability or verification reasons. Depending on business need this mode could also be a powerful machine capable of running applications on a permanent basis. Volume: Unit of backup, a single file system comprising many files and directories that are backed up at the block level. END TO END PROTECTION WITH ENTERPRISE IMAGE DESTINATIONS

Enterprise Images Destinations are a part of the APM (Advanced Protection Manager) suite of products. This feature is implemented entirely in software and once installed on a node would allow that node to function as a neaiiine destination for application aware Forever Image Ihcrementals. This EID node could be configured in various ways (local disks, iSCSI storage etc.) to offer various degrees of protection and reliability. Image backups from various nodes would be consolidated, nearlined and versioned on this device. Instant Availability for file-systems and applications would be leveraged off these versioned images.

Fig. 1 shows a typical enterprise deployment of an embodiment of the invention, showing a secondary storage server 107 utilizing inexpensive SATA disk drives, connected in turn to further arrays of servers 103, NAS device 104, a remote secondary storage device 105 and tape storage 106. This backup arrangement is used

to remotely manage backup and recovery for networks comprising both small (101) and large (102) remote sites. Block level backup clients are used to perform block level backup operations where indicated (111, 112, 114, 115). Replication to tertiary storage 113 (wherein secondary storage server 107 also serves as a tertiary storage) and tape 116 (to tape drive 106) are also shown. The various elements and backup steps shown in Fig. 1 will be further discussed in the sections of this disclosure that follow.

Architecture: Basic: The EID node would have locally attached SATA drives configured as hot- pluggable RAID5. This storage would be used as a repository for images. Versioning would be implemented via snapshots available on the system (VSS for Win2003 or LVM/EVMS for Linux). Images would be exported as read-write LUNs via bundled iSCSI target software. Thin:

The EID node would only have a small local drive (ideally mirrored) to hold the OS and EID software. A back-end iSCSI storage array (or similar network intelligence) would be used as the actual destination for backup images. A storage array would necessarily need to expose LUN creation, snapshot creation, LUN cloning, LUN masking/un-maskmg features to be a candidate for participation in a bundled EID solution. VSS/VDS or SMI-S APIs may be used to standardize on the

interface between EID software and external storage. Thin Shared: This is a variation of the above where the networked storage array is shared between the source machine(s) and the EID node. Backups can be optimized in this configuration by sharing a snapshot between the source and destination. The EID node would act as a backup head in this configuration. EID with Double Protection:

Backups need to be protected via further backups to tape or disk. This is termed Double Protection. (Refer to Double Protection document) First backups to disk on EID nodes could go to tape devices on the SAN or other disk distinct from the storage where the first backups reside. This would be second or third tier storage residing on the SAN or attached to some remote appliance (possibly another EID

node). Thus EID is the key enabler for an End-to-End solution for data protection based on multi-tiered storage.

Configuration:

APM client node: These nodes would be configured with the APM client and support for multiple snapshot providers (if available). The APM client would be capable of backing up EID targets to secondary storage, which can be to vendor-supplied storage hardware or generic ATA storage hardware. The snapshot support could be basic (bundled EOFM) or complex - each volume may have a separate snapshot provider. (When multiple snapshot providers are present, their use must be pre-configured or indicated by the EID node) Application support when implemented is available simultaneously for both secondary and EID targets.

APM Server - EID node:

This node would have the EID software installed with a storage specific plug- in depending on the back-end iSCSI storage (if any). The plugin configuration would be hardwired during installation along with licensing information. The basic configuration would be supported on two different set of commodity OSs - Windows 2003/NTFS and Linux 2.6 with ext3fs/xfs with LVM/EVMS. The requirement essentially is 64-bit journaling file-system with sparse file support with multiple persistent snapshots. Any system meeting these criteria could be a candidate for an EID node. (Additional properties of the file-system like compression and/or encryption although not essential could be employed to provide additional features at additional complexity and/or overhead) Backup Flow: Fig. 2 schematically illustrates the creation of a point-in-time snapshot, block- level incremental backup, and point-in-time full volume image, as well as a file-level restore operation.

Snapshot Phase:

The Protocol director contacts APPH (Application Helper, which mediates application (SQL Server, Exchange, etc.)-specific interaction at the beginning and end of backup) with BACKUP_PREPARE. APPH contacts the Snapshot Handler, which encapsulates snapshot code and incremental block tracking interfaces, to snapshot a set of volumes and flush change journals. The Snapshot Handler would do DISCOVERJLUNS as part of file system discovery. On detecting that some LUNs

are back-ended by supported iSCSI (or FCP (Fibre Channel Protocol)) vendors it would invoke a vendor specific method to take a snapshot of the set of volumes that reside on the same iSCSI storage entity (for example a volume containing a set of LUNs on a storage device). A specialized provider would exist per storage vendor providing this functionality or VSS or SMI-S providers could be used if available from the storage vendor. Additional configuration information will be required per back-end storage node to provide this functionality, which would have to be obtained from the database. (This information may be cached or saved as part of a local configuration file.) Since most external providers would not provide change journal support both external (or VSS mediated), a bundled EOFM snapshot would need to be taken. The EOFM snapshot would solely be used for flushing the change journal and tracking changed blocks. The external snapshot would represent the real backup instance or a consistent source for remote copy. The EOFM snapshot needs to be taken first, followed by the external snapshot to produce a consistent image. A small window exists between both snapshots where blocks may change. Since applications are already quiesced (application state has been mediated via APPH so that the application knows that backup has started and has flushed its transactions to disk) no I/O should be generated for them. No file-system meta-data should change either (File systems are capable of recovering to a crash consistent state at any event). An individual file may have blocks changed which would not be captured till the next incremental. Note that the window is small and the odds of an unsupported application having an inconsistent state are extremely small.

APPH would at the end of the process create a content file for the backup specification. This file will be augmented with

vendor specific info with possibly a logical name and a persistent snapshot id along with a local snapshot volume created by EOFM, VSS or third-party provider. Data Transfer:

SVH contacts the EID software with a CREATE_RELATIONSHIP message (for the first backup) and passes the content file as the source path. The EID software on the EID node then establishes connection with corresponding software ("Node software") on the source node and passes the content file path. The Node software on the source side then reads and passes the contents of the content file back to EID software on the EID node.

Variation I: Shared Snapshot=Backup

The EID software examines the vendor specific snapshot info and determines whether the vendor is supported and licensed. If the answer is yes the EID software tries to determine via local query snapshots existing on the shared storage device and if it determines the shared snapshot can be used as backup then the process completes. The allocation bitmap is also obtained at this point. The EID software stores the relationship, a combination of the source node+source drive(or unique id)+destination node+lun name in its local database. The allocation bitmap is also saved indexed by snapshot id. Snapshot on the EID node:

The CREATE_SNAPSHOT from SVH returns with the shared snapshot in the previous step.

Error Recovery: Not needed for this scenario. Restart after Cancel:

Not required as the backup should be very quick. File History:

File history is generated (optionally) on the EID node using the backup LUN. The File history is to be conveyed to Backup Express Master server in some implementation specific way. Incremental Backups:

These proceed in the same way as base backups except for the fact that the change journal is passed in its entirety to the EID node, which then stores the CJ in its local database indexed by the snapshot id. Checksums:

Checksums may be calculated for all allocated blocks on the LUN image and saved away in the EID database indexed by snapshot id. Checksums are important for three reasons:

1. Ability to verify after write. 2. Aid in reliable check-point re-start.

3. Ability (albeit at increased cost) to do incremental backup with block level tracking.

APPS:

The APPS volume comprises files generated on the live file-system after the snapshot is taken and as part of the POST B ACKUP event. These files do not exist in the shared snapshot. These files need to be independently backed up. Variation II 'Local Copy To Backup LUN' has to be used in this case. Although APPS appears as a virtual volume, the backup of APPS is effected by copying whole files (file by file backup) and not volume-oriented block copy. Variation II: Local Copy To Backup LUN

If the EID software determines that the shared snapshot cannot be used, it creates a backup LUN on the iSCSI storage or locally, naming it uniquely with the source node + drive id combination. The hostname+portid+targetname+lunid is returned to the source EID software as part of the initial handshake.

The source side Node software then calls MAP_LUN (which indirectly uses iSCSI login) with the information passed from the EID node. MAP_LUN exposes a device mapped to the local namespace. The Node software begins to copy allocated blocks from the local snapshot of the device to the iSCSI-mapped device. During this process it passes status/checksums/progress to the EED Software via the already established channel.

Snapshot on the EID node: The EID software takes a snapshot of the backup LUN or some covering entity and returns the snapshot id.

Error Recovery:

Should not be needed since iSCSI connections for data transfer are reliable and have built in recovery and error connection. The EED software should be able to recover from errors on the control connection transparent to DMA.

Restart after Cancel:

This needs to be implemented. The EED software needs to remember the last successful block written and pass this on during the initial handshake indicating that this is part of re-starting an aborted backup. File History:

File history is generated (optionally) on the EID node using the backup LUN. The File history is to be conveyed to Backup Express Master server in some implementation specific way.

Incremental Backups:

These proceed in the same way as base backups except for the fact the change journal is used locally to copy only changed blocks on to the backup LUN as part of the process. Checksums:

Checksums may be calculated for all allocated blocks on the LUN image and saved away in the EID database indexed by snapshot id. APPS:

The APPS volume comprises files generated on the live file-system after the snapshot is taken and as part of the POST_BACKUP event. These files do not exist in the backup snapshot. After the APPS LUN has been mapped locally, it has to be formatted as a locally recognized file system. Then APPS directories/files are copied whole (file by file) from APPH directed locations (and not from a snapshot) onto the APPS backup LUN. During incremental backups the APPS LUN has to be cleared and a new set of APPS files copied. (The older snapshots would retain the previous versions of APPS files) Variation III: Network Copy

Like Variation II if the EED software determines that the shared snapshot cannot be used, it creates a backup LUN on the iSCSI storage or locally, naming it uniquely with the source node + drive id combination. LUN creation may fail if it is not supported on this node (really basic configuration) If this happens hostname+portid+targetname+lunid is not returned to the source Node software as part of the initial handshake and Variation III is indicated.

If Variation III is indicated or there is no iSCSI or other means of LUN mapping support on the source node then source side Node software begins to read allocated blocks from the local snapshot of the device and send it across the network to the destination EID software. The destination EID software reads from the channel and writes out a sparse file on some pre-defined volume on the destination. Either end in this process may generate checksums. Snapshot on the EID node:

The EID software takes a snapshot of the volume containing the backup image file and returns the snapshot id to the DMA.

Error Recovery:

Needed to recover from network outages via checkpoints kept on the destination.

Restart/Restart after Cancel: This needs to be implemented. The EID software needs to remember the last successful block written and pass this on during the initial handshake indicating that this is part of re-starting an aborted backup.

File History: File history is generated (optionally) on the EID node using the backup image. Incremental Backups:

These proceed in the same way as base backups except for the fact the change journal is used locally to read only changed blocks and then transfer them over the network on to update the backup image on the destination.

Checksums: Checksums may be calculated for all allocated/changed blocks on the backup image and saved away in the EID database indexed by snapshot id. APPS:

Then APPS directories/files are read whole (file by file) from APPH directed locations (and not from a snapshot) and copied across the network to the destination EID software where a directory structure (under a pre-determined backup directory location) is created to reflect an identical copy of the files at the source. During incremental backups the APPS directory has to be cleared and a new set of APPS files transferred and re-created from the source. (The older snapshots would retain the previous versions of APPS files) Plug-in Architecture for External LUN/Snapshot management:

EID backups depend on snapshot creation, LUN creation, LUN cloning etc. Both the source side and the EID side of the backup process are consumers of these services. To facilitate easy architectural separation and be able to plug-in various vendors an interface with an associated vendor specific provider (in the form of a DLL or a shared library) needs to be implemented. The default implementation would use the bundled iSCSI provider on the EID node, but could be replaced by a vendor specific implementation if warranted. The interface would provide generic LUN creation/deletion, LUN cloning, snapshot creation/deletion functionality. An augmented version of the interface might add functionality for block level mirroring

and other salient features (for example: a Secondary to Tertiary Replication feature), which may be taken advantage of for supporting efficient/elegant Double Protection methodology. EID database: A small database on the EID node is needed to maintain configuration (like back-end iSCSI storage), licensing, snapshot ids, checksum info etc. This would be especially necessary where the EID node is back-ending some iSCSI/shared SAN storage. Backup Express infrastructure would be dealing with a unique snapshot-id, but the EID software has to translate this to an exact network entity by de-referencing the snapshot-id via the local database.

A simple implementation may be a set of directories named with snapshot ids containing block allocation bitmaps, incremental bitmaps, checksums, file history etc. Double Protection to Tape:

This will be done via a regular NDMP (Network Data Management Protocol) backup re-directed to job handler from SSSVH. (Refer to the separate discussion Double Protection) The important thing to note about DP to Tape is that a full/complete image of a first backup is created on tape. Subsequent tape backups are full copies of other first backup instances. No notion of incrementals or in any other way relating one tape backup image to another is part of this design. Double Protection to Disk:

Double Protection to disk (DP2D) prolongs the life of a backup image on disk further by creating another backup on disk of the original/first backup. Every effort is made in this case to create subsequent backups by transferring incremental data to update tertiary backups. Various scenarios are: Multi-tiered storage visible to EID node:

In this scenario the tertiary disk storage is accessible from the EID node (Secondary and Tertiary storage may be part of a large multi-tiered storage deployment accessed via a uniform single vendor interface - Hitachi TagmaStore). DP backup in this case would proceed via a local block-level incremental copy performed by the EID software after the appropriate tertiary location is selected and a LUN un-masked/mounted on the local EID node.

Block Mirroring between Single Vendor Nodes: In the case a vendor has an efficient and appliance implemented block mirroring method for transferring data between secondary and tertiary nodes, the EID

software would trigger and image transfer/update via vendor specific API set to create a Double Protection backup.

EID Node to EID Node:

When tertiary storage is physically separated from the EID node, the remote EID node would initiate the backup via "Network Copy" to pull data from the local EID node.

EID Node to Secondary:

When data has to be transferred between an EID node and a secondary node, the applicable Backup Client transfer method would be used, i.e. the secondary would be contacted and asked to pull data from the EID node. The EID Software would recognize a DP2D backup and update the secondary image from appropriate (usually latest) snapshot using saved bitmaps.

Backup Mechanism:

Once a double protection job is created for protecting first backups, the Protocol director initiates an EID backup, much like a regular EID backup except that the snapshot phase is skipped.

A CREATE_RELATIONSHIP is sent to the destination EID software (this could be an EID node co-located with the destination, a remote EID node, or another type of secondary). If the EID software detects that it is the source node for the backup, it uses appropriate mechanism to either copy the image locally (using allocated or incremental bitmaps saved with the backup) to a tertiary destination or invoke a vendor specific method to affect this transfer. If the EID software detects that the source is remote it initiates a regular EID backup using a previously described mechanism. The backup is saved on the destination EID node like a regular EID backup, implying that this process can be cascaded indefinitely.

The snapshot-id, which comes back from the NOTIFY response to secondary snapshot creation, is cataloged as part of the DP backup and linked with the original first backup. (For detailed explanation see the separate discussion of Double Protection.) Restore from Double Protection Backups:

Refer to description of Double Protection. Restore Browse:

When file history is generated at the end of backup on the EID node and incorporated into the Backup Express database, browsing happens normally via the

catalog browse function of the dB. When file history is not generated (when generating file history is computationally intensive or would require too much storage) the NDMP Directory Browse function may be used by contacting the ELD software. Browsing may be provided by mounting the backup LUN on the EID node and then browsing the file-system using existing 'snap dir list' mechanism or by generating 'rawtoc' from the image file when browsing is necessary. Double Protection to tape requires that file history be generated during a Double Protection operation either as part of the image or to construct a file-by-file archival format if the option to mount the LUN as a recognizable file system is not available. Restore Flow:

Directory/File Restore:

Once the restore selection has been generated (either by the user or by the Protocol director after the backup document for the instance has been translated by APPH from application objects to files) and a content file has been created SSSVH contacts the Node software on the restore target passing it the content file, which EID node to get the data from, path and snapshot id on that node. The Node software on the restore target then contacts the EID software passing it the restore path and the snapshot id. Once the EID node examines this information it makes a determination of whether the snapshot-id & volume combination can be exposed as a LUN on the restore target. If this is possible (much like backup) a LUN is created by the EID node, either locally or on shared SAN storage and hostname+portid+targetname+lunid is passed to the restore target. (Note: hostname may not be the same as the EID node) Once the Node software on the restore target is able to map this LUN the handshake completes. For Instant Availability this essentially completes the restore process. Otherwise the Node software does a local copy of files/directories from the mapped LUN to the restore target locations. (Note: This is exactly like how APPS files are logically backed up) Fallback- It is possible that EID node determines that LUNs cannot be exposed to the requesting node (e.g., for security reasons) or that after the initial handshake completes the requesting node cannot map the LUN. In this situation (a low priority) a traditional restore proceeds where the EID software reads the requested files from ' the backup image and sends them over the network and the Node software on the restore target recreates the file locally from the received data. In this situation

'rawtoc' is required, either pre-existing from a post backup process or created on the fly for restore (and then cached if desired).

Error Recoveiy/Restartability:

This is unnecessary for LUN mapped/IA style restores but may be useful for traditional restores (if that is implemented at all) Instant Availability Restore:

As in other block-level restores MAP_LUNS will be called (as implemented in the Snapshot Handler) to map a set of volumes via iSCSI or FCP on the restore target from the selected snapshot. The Snapshot Handler will call CREATE_LUN_FROM_LUN on the EID node to create and expose a LUN within a snapshot. The APPS volume will then be similarly mapped to the local namespace either via a local iSCSI mount or a network mount. Once this step completes SSSVH will direct APPH to complete the restore. APPH will copy log files if necessary from the APPS volume to the IA

volumes to recover the application or database. Note that the EID software is not contacted for IA restores at all.

The backup data transmitted across the network as part of a differential block level image has a disk signature attached to beginning which has the appropriate information to virtualize the backup of a volume as a whole SCSI disk with a single valid partition. During restore this read-only image is transformed into an iSCSI addressable read-write LUN by creating a sparse file backed by the image within the snapshot. This LUN file is persistent and can function as primary storage aggregating changes as well as original unchanged data from the backup image. The LUN can both be mounted as a stand-alone disk or part of a RAID set. Error Recovery/Restartability:

N/A.

Restore via Local Volume Rollback:

Volume rollback is only possible if restore happens to original location and all change journals since the time of backup exist. If these criteria are not made a full volume restore can be triggered (this is a de-generate case of volume rollback anyway) or the restore job fails. (Given the functionality of IA restores this may not need to be implemented at all.)

An option indicates that volume rollback is desired, in which case a VOLUME_ROLLB ACK message is sent by the Protocol director to the Snapshot

Handler (much like IVLAPJLUN). This message contains the backup jobid (which uniquely identifies the point-in-time of the backup) and the volume in question. If volume rollback is possible the Snapshot Handler locks and dismounts (applications hosted by the volume are shut down or off-lined by APPH) the volume in question and then takes a snapshot to flush the change journal. All change journals since the time of the snapshot that is being restored to, are logical- ANDed to create a bitmap file which is returned (the file name only) to the Protocol director. The Protocol director adds the bitmap file to the content file and passes this on to the EID software, which uses the bitmap file to restore only a set of blocks from the mapped LUN or across the network.

If traditional full volume restore is implemented then the allocation bitmap has to be passed to the Node software on the restore target from the EID node so that only the allocated blocks are copied. If network copy is used the EID node already knows which blocks to send. After restore completes the volume is unlocked and re-mapped in the local namespace and applications/databases re-started and on-lined.

Restore via Volume Rollback in a Thin Shared Configuration: This mode of restore requires back-end storage support of single file or LUN rollback. Volume locking and application shutdown happens on the restore target node mediated by the Snapshot Handler and APPH exactly like above.

During the initial handshake for Volume Rollback the restore target passes covering information for the target volume, (for example: D:=filerA,vol3/lun2) to the EID software. The EID software on determining that the back-end storage supports this feature and that the snapshot and the restore target LUN are logically related calls a back-end API (part of the plug-in interface) with two arguments - the snapshot that is the being restored from and the target logical entity or LUN that back-ends the volume on the restore target node.

Volume rollback on the back-end storage happens asynchronously and may take a while depending on the divergence between the live file- system and the snapshot (but should be quick since only local copy is involved). Once this completes the restore ends and applications can be re-started. (An example of this scenario is a single file LUN snapshot revert on an NAS device.)

Error Recovery/Restartability:

Full Volume Restores: Only important for large foil volume restores. May be implemented by a restart mechanism similar to backup but with the checkpoint tracked by restore target Node software and communicated on a re-connect. Whether restore needs to be re-started after cancel by the DMA is outside the scope of this document.

Local Volume Rollback:

Error recovery should be un-necessary since the restore involves local copy. Re-startability after cancel/suspend may be desirable. Application Supported Volume Rollback: Error recovery should be unnecessary but re-startability should be implemented if the back-end storage supports restarts.

ExpressDR Restore:

This is a special case of full volume restore where the restore target is running Linux. The Linux Node software may be driven by a modified version of jndmpc to work exactly like above, taking advantage of an iSCSI initiator if available on the custom Linux kernel. Error Recovery/Restartability would be essential in this situation. Additionally a standard mechanism needs to exist for browsing snapshots for ExpressDR backups of a given node. This should be part on an interface exposed by the EID software or the Snapshotg Handler on the EE) Node. A snapshot directory listing may be sufficient with a pre-defined naming convention for snapshots, or a suitable interface may need to be defined for enumerating matching snapshots.

Error Recovery/Restartability: This is very desirable for large restores and should be implemented in similar to full volume restores. Security/Virtualization/Compliance/Self Provisioned Restore:

Nearlined data needs to be more secure than data on offline media (like tape) since data is live and accessible over the network given proper permissions or if a small set of accounts are compromised. One option would be to encrypt data that resides on nearline storage (Native file-system encryption could be used if available). This would slow down Instant Availability Restores but the added security may make it worthwhile. Double Protection to disk and/or tape, especially if they are for long term archival reasons are also prime candidates for encryption.

A few user accounts (Backup Express admin and root or Administrator on the EID node) protecting backups of a lot of machines consolidated on a single EID node, may not be secure enough for most enterprises. Multiple admins each having responsibilities/rights over a set of backup images maybe more acceptable (In this situation the super-user would not necessarily have rights over all backup images). Some style of RBAC (Role based access control) may be implemented by using existing security mechanism on Windows 2003 or Linux 2.6.

Since complete images of application servers are stored as backup images on the EID node, these set of images (at various discrete points of time in the past) are a prime candidate for virtualization. Each client node or application server can be virtualized as it appeared at some point-in-time in the past using some off-the shelf or OS dependent virtualization software. The potential for secure virtualization of machine states (where only authorized persons have access to machine data) allows enterprises to implement just-in-time virtualization for administrator-less restores, compliance, analysis or other business salient reasons.

Regulation compliance or litigation discovery are important applications of the EBD paradigm where data on the EID node could be virtualized to some point-in-time in the past for compliance inspection at very little additional cost. Double Protection to disk or tape targeted at specialized compliance appliances like secondary WORM storage or WORM tapes enable an end-to-end solution starting from backup, to near- term restore and long-term archival to meet compliance requirement.

Self Provisioned Restore refers to administrator-less data recovery where end users typically restore files without help-desk or administrator mediation. This is possible as data is stored on the EID node preserving original file-system security. Once Instant Availability or other techniques are used to map volumes back to some well known location users can find and restore data using existing and familiar tools. (The Backup Express GUI may also be used to find and restore data without having to login as an administrator.) An intrinsic property of the EID architecture enables self- provisioned end-user restore and thus reduces TCO (Total Cost of Ownership) significantly. Example:

Fig. 3 shows block-level incremental backup and file-level incremental restore operations in greater detail than Fig. 2, in a manner that illustrates a number of the

foregoing principles. The example shown involves the following events and operations:

• 2:00 a.m. A base backup is performed of primary system 300 during an early a.m. backup window. Note that only allocated blocks (301) are backed up. The unallocated blocks (320) are not transferred to the secondary storage unit

330, reducing elapsed time and secondary storage requirements. The snapshot (341) on the secondary represents all the data (volume/directories/files) on the primary at 2:00 a.m.

• 10:00 a.m. This is an incremental backup, since all backups after the base backup are automatically incremental. Note that only the blocks that have changed (302) since the base backup are transferred. The snapshot (342) on the secondary is a synthesized base backup image that represents all the data (volume, directories, files) on the primary at 10:00 a.m.

• 11:00 a.m. Only blocks that have changed (303) since the 10:00 a.m. backup are transferred. The snapshot on the secondary (343) represents all the data on the primary at 11 :00 a.m.

• 12:00 p.m. The 11 :00 a.m. snapshot (343) is selected from the backup instances (snapshots) displayed on the Backup Express restore screen. From this backup instance, three files (351) are selected for restore.

DOUBLE PROTECTION

Double Protection protects first image backups to intelligent disk storage by backing them up to tape or disk, managing their life-cycle and providing direct restore from tape when first backups have expired or disk storage is unavailable. APM TO DISK (APM2D): First Backups:

1. Images of file systems are backed up to disk along with application specific meta-data (APPS). This data resides in a form that enables Instant Availability and/or Instant Virtualization.

2. File systems/OSs for which image backup is not supported are backed up to disk as files and reside under a destination directory as a point-in-time copy of the source file system.

Double Protection explained:

Double Protection creates at least one (and as many as desired) virtual copy of the first backup to disk or tape. The crucial point here is that subsequent backups are identical untransformed copies. Since the first backups are frozen point-in-time images, copies can be made at any time in the future and still capture the original state of the file system. Twinning is not needed anymore since as many copies of an original backup can be made as soon as or whenever policy dictates. For supported applications, application consistent snapshots are saved to tape as if the tape backup was done at the time of the original first backup. Presentation/Scheduling:

The GUI would present in a Double Protection screen a list of first backup jobs, which are candidates for double protection. This would look like a traditional image/or NDMP backup screen except for the fact that the left pane would be backup jobs. (Device selection may be avoided initially by implicitly selecting default cluster and mediapool for the containing nodegroup). The DP job would be saved as a NDMP job with the first backup jobname or a first backup jobid as part of the definition. The schedule would be simple -just a backup schedule like APM2D, no base incremental or differentials settings. DP jobs with a specific instance selected of a first backup job (i.e. jobid) would have no associated schedule and the job would be deleted after it is run. When job handler receives JOB_START and determines that this is a DP job would issue a CREATE_DP_JOB to the database specifying job name or job id as argument. The dB can obtain given the jobid (and by looking up the snapid) the backup document for the job. Given a job name the latest backup job id would be used to find the backup document for the job. The backup document contains the entire state of the first backup needed to be able to construct an NDMP job to tape identical to the original APM2D job. A one-to-one mapping of tasks in the original would be created in the DP_JOB resulting in an equal set of source statements.

For example a APM2D job with tasks C:, D:, APPS: would be translated to three tasks

/vol/vol 1 /. snapshot/snapname/qtree 1 , /vol/voll/.snapshot/snapname/qtree2, and /vol/voll/.snapshot/snapname/APPS-qtree.

CREATE_DP_JOB would return a temporary job name whose definition once obtained by job handler would allow the NDMP job to proceed. Once this job creates a copy to tape it would be as if a backup to tape was run at the original time of the disk backup. The first backup jobid and taskids are needed for co-relating the DP jobs tasks with respect to the first backup. As part of CREATE_DP_JOB dB could pre-catalog the DP job creating catalog entries, which would be validated if an actual TASK_CATALOG came in.

The CREATE_DP_JOB could also be called by SVH when a necessary condition is triggered (running out of snapshots etc.). SVH could then run this job via JOB_START etc. following a backup or even before a backup.

Comprehensive scheduling incorporating both disk and tape and life-cycle management is outside the scope of this project and would be considered at a later stage. Running DP j obs :

Double Protection jobs are APM backups mediated via the EID software or external NDMP data servers (including proprietary NAS backup methods). The first backups could be image files or replicated directories. When the EID software backs these up it would recognize that DP backups are being made and back them up preserving original format if image or as logical backups if they are replicated directories. External agents would back up images or replicated directories in their native format (dump or tar).

In the event DP backups go to tape the legacy job handler path would be used. DP backups directed towards tertiary disks (Secondary to Tertiary Replication) would be handled by SSSVH or by some external agent (may involve simple scripts followed by a cataloging utility step)

In all cases no file history would be generated or captured since the identical file history for first backups makes this redundant.

All restores would be done via the Node software, regardless of originating format. (This would mean understanding external dump or tar format as needed.) Archival Format/Compliance:

For long term archival or regulation needs DP backups may transform image backups to logical backups in some portable format like tar, cpio, or pax. These backups could go to WORM tapes or WORM drive to meet compliance requirement.

Data would be restorable from this archive using file history saved during first backups. Direct Access Restore (DAR) would require re-saving file history with associated fh_info thus requiring file history generation during the double protection process. Generally available utilities like tar etc. could be used to restore files from archival formats independent of Backup Express. The present design provides freedom to make and/or publish different archival formats. Cataloging:

Each DP job would catalog as many tasks as the original backup in 'sscaf. New fields in sscat for original task and job ids would be added to track reference to the original job. (As part of this we could also add a snapid filed as part of sscat since this is a high level and crucial construct for first backup jobs) The DP jobs would have their own equivalent disk entries in sscat with path names reflecting secondary disk locations. Example sscat (partial columns):

Catalog Condensation and Job Expiration:

Since the first backups and subsequent DP backups are treated as separate jobs, each would have their own retention period. As first backups expire checks would be made to ensure that DP backups exist depending on policy. A warning may be issued or a DP job may be triggered at this point if a determination is made that there are unprotected first backups.

During condensation of primary jobs the catalog entries for the first backup would be retained and not deleted to preserve file history. The backup document would also be retained since this is necessary for application restore. The original job id is always retained as part of the promoted job, since this is what needs to be reflected a part of the restore browse. If multiple DP j obs exist for a given first backup they all contain the original job id, which would point to the original ssfile.

This process should be relatively simple since a single pass through the catalog table would be all that is required during condensation. Restore Definition Generation: Restore browse would return the SNDMPDAT A from the original job instance for restore presentation. The RJI process would also be enhanced to include file history from the original ssfile to create a proper restore specification. The process would involve producing the tape windows involved in the DP backup along with the restore path names from the original ssfile. The root directories (the only thing cataloged) in ssfile for the DP backup would be ignored. Restores: Fault tolerant/Location independent

DP tape backups being regular NDMP backups would show up under regular NDMP restores and can be used to restore directly to any compatible file system. In situations where the original secondary disk location is destroyed or corrupted these backups can be restored to original location to either recreate APM2D locations or to stage restores or effect

virtualization. These restores can be handled by job handler as normal NDMP restores and can be part of a complete solution if no applications are involved.

A disaster recovery or full node backup of the secondary disk node is treated as a separate backup and may be used independently to restore the secondary in case of disaster.

The APM2D restore view would be unchanged, except for the fact that if DP backups exist for first backups they would not be displayed. For expired backups if DP backups exist they would show up and be presented as nearlined backups. The restore browse process would need to be augmented to return NDMP backup instances as APM2D backups. The restore selection would be passed on to SSSVH as today. (It is possible to create a NDMP restore job for application restore if job handler implements the restore side of APPH processing but this may be limited in terms of handling fault tolerance well.)

After APPH has been contacted for application restore and the restore file list determined the Protocol director would try to cycle through available disk destinations in order to satisfy the restore selection. If this fails (first backups have expired or disk destinations are unreachable) a NDMP restore job from tape would be constructed and run via JOBJST ART (presumably run by job handler). Once this successfully completes APPH will again be contacted and the restore completed.

"LAZY MIRRORING"

A primary volume may be mirrored onto a secondary volume in accordance with the following procedure: Mount the primary volume

Mount the secondary volume

Create a list of blocks to be copied from the primary volume to the secondary volume.

Write new blocks to both the primary and secondary volumes as they arrive

As blocks are written, remove those blocks from said list of blocks. Traverse said list, and whenever bandwidth is available and convenient, copy blocks encountered as a result of such traversal from the primary volume to the secondary volume. Continue until all blocks on said list have been copied.

The end result of the foregoing is that the secondary volume will be synchronized with the primary volume. This technique does not require stopping processing on the primary volume, nor does it impose any constraints on how much time can be taken to complete the copying process. The "lazy mirroring" technique may be used, for example, to restore a physical primary device after an "instantly available" virtual device has been utilized, for example, after the failure of a primary device. The virtual device will be used temporarily, in that the data on it will be intact as of the point-in-time of its snapshot. However, the virtual device may be only a temporary solution, and the business will need to restore to a replacement primary device as soon as is feasible. "Lazy

Mirroring" provides this capability in a manner that allows processing to continue uninterrupted, and allows the actual copying to proceed at its own pace while minimizing the load on other system components.

The "lazy mirroring" technique may also be advantageously used to "resilver" a mirror that has crashed or gone out of sync, while the primary mirror remains in production.

Moreover, the "lazy mirror" technique may be used anywhere where it is desired to copy a volume without stopping it, and to do so without engaging in extraordinary measures to save time.

ELIMINATING REDUNDANCY IN BACKUPS AND FILE SYSTEMS

Where a plurality of systems are being backed up in a backup operation, it is not uncommon that machines will have a large number of blocks that are identical to blocks on other machines involved in the backup. This may arise when multiple machines have installed on them the same operating system files, or the same applications or data files. It is redundant to store blocks having identical content multiple times. The redundancy concerns not only the redundant use of storage, but

also the redundant use of bandwidth in transferring and storing the duplicate blocks. Furthermore, even in a single file system it is not uncommon to have duplicate blocks as a result of duplication of files. This represents a redundancy as well.

Such redundancy may be eliminated in a backup context by taking a digest of every block written to the backup data set, and putting the digest data in a list or database. Comparison of block digests is preferably performed on the server side.

If a node to be backed up has a large number of blocks that have changed and need to be backed up, it sends a list of those blocks with their digests to the backup server (it may also be the case that the node has created in advance lists of block digests for some other purpose, such as determining which of its own blocks have changed, such that those digests do not have to involve a separate step to create them).

The server then compares the block digests and requests those blocks for backup, which it has determined it does not already have (the list or database of blocks is stored in such a way as to facilitate rapid lookup using the digest as a key). The complete list of blocks sent by the remote node is saved (including those sent over plus those that the server determined it already had), as part of the backup catalog.

Preferably, if the node being backed up has only a small number of changed blocks, it simply sends them in that circumstance and skips the redundancy check.

A similar technique is employed for eliminating redundancy in a single file system. Each block to be written to the file system is digested, and compared against the digest of the blocks already stored (here again, the list or database of blocks is stored in such a way as to facilitate rapid lookup using the digest as a key). If the identical content block already exists on file system, the existing directory point is used and the duplicate block is not written. When a file are deleted, its blocks are deallocated from that file. If other files use the same block, those allocations remain in effect (a block is not "free" until no files reference it).

EXAMPLES: FAST APPLICATION RECOVERY The following are a series of examples illustrating Fast Application Recovery as provided by the present invention. Introduction to Examples:

The examples illustrate the ability provided by the present invention to bring an application on-line quickly on a stand-by or original server by attaching to virtual storage created out of backup images on a filer, such as a NAS filer.

Consistent volume images from source nodes are nearlined with their associated application consistent state as backups, typically on NAS Filers. Users deal with application logical objects while the Backup Express agent creates hot base backups of physical objects that comprise the application. Forever Incremental Images ensure that only blocks changed since the last backup is copied to the filer without sacrificing the fact that all database backups are full. Since the application data and state is nearlined restore is affected very quickly by recovering a point in time copy of the application files, then bringing the application online and applying a small number of redo-log records. FAR recreates storage as it existed at the time of backup, establishing the physical relationships that the application logically expects and then recovering the application to a fully functional instance. Mechanisms Illustrated:

Application restore is broadly a two step process: data file(s) need to be restored followed by application recovery (sometimes known as roll-forward recovery). The user selects an instance of backup or a PIT image (usually latest) depending on nature of disaster, type of user error or other business need. The first step is accomplished by creating addressable virtual storage (LUNs) on the fly on the filer from the user selected PIT volume images. These LUNs are then made visible to the target node in question. These are then attached as local disks on the restore target

via iSCSI login to the filer. This process is near instantaneous since no actual data movement is involved. Once application data files are visible in the local namespace of the target node, applications are then programmatically recovered using appropriate application specific API. This may require application of additional log files which are obtained as necessary from the filer backup location. This brings the application instance up to the point in time of the backup. If current logs are available then roll- forward to the point of failure is possible. Since the backup was a snapshot backup, the application was in hot-backup mode for a very short time, only a few transactions need be applied to bring the database to a consistent state. The relative simplicity and quickness of these steps enable the application to come up in a matter of minutes after the FAR process is initiated. Compared with traditional restore FAR is orders of magnitude faster reducing application downtime from days or hours to minutes. FAR scales independently of the size of the data set. Post-Restore: FAR is not the end of the story. As FAR completes block change tracking may be enabled and local slice attachment may

be done if needed. This enables background restore to proceed while the application is up and running. Incremental backups may be started from the point in time of restore since tracking of changed blocks is enabled. The application may eventually fail back to the original or another node with minimum downtime with all recent changes (since restore) preserved. Requirements:

• Source and target nodes need to be running and licensed for the APM. (Applications if any may need to be licensed separately.)

• The NAS device or secondary storage unit needs to be licensed for iSCSI. • Target nodes need iSCSI initiators software installed (iSCSI HBAs are also supported)

• Stand-by nodes need to be pre-configured with a minimal application install.

• Platform/application support includes Windows XP/Wmdows2000/Windows2003 and SQL Server 2000 (SP2+), Exchange 2000 (SP2+) / Exchange 2003, SQL Server 2005, Oracle and Linux.

Various scenarios and applications for rapid application restore and the lifecycle of data following restore are explored in the following sections:

Example 1. IV (Instant Verification) for APPs

Need: Restore is always a shot in the dark since backups are never really verified. Tapes are unreliable. Verification usually amounts to verifying internal consistency of the backup image. Application consistency and recoverability is a matter of chance.

Approach: IV for APPs verifies application backups near-instantly by restoring (FAR) to an alternate verification node or the original node when possible.

The application is then recovered to complete the process. This can be scheduled so that every backup is always checked for integrity and no additional fire-drills need to be performed for recreating disaster scenarios.

PIT Image Used: Usually latest but could be images from the past if verification is batched.

Where Performed: Usually done on an alternate node where a minimal application installation is pre-created. The same node as the source for backup may be used if the application supports it. (For example: Exchange 2003 configured with Recovery Storage Group or SQL Server with the option of renaming the database being verified) Verification on the original node is usually not recommended since this places extra stress on the application server.

Modes: Lightweight Verification: The application (usually database) restarts/recovers correctly thus verifying correctness of backup.

Comprehensive: If necessary further verification can be performed (more resource intensive) using application specific techniques to verify that all database pages are clean and/or logical objects function properly. (Imagine a database query which spans tables and the result is a clear vindication of database health) Application Specific Notes:

Exchange: Mounting stores are usually a significant step. Further verification can be done using 'eseutil' on an alternate node.

SQL Server: Mounting databases are usually a significant verification step. Further verification can be done via 'DBCC or by running SQL queries. Follow-Up:

None. Verification is a transient operation and an iSCSI logoff or reboot will clear the machine state. IV for APPs may be configured so that the next verification run will clean up previous verification mappings. The machine state with mapped

drives need not be preserved and thus no further backups are necessary of this alternate node.

Example 2. IA (Instant Availability) for APPs for Business Continuity

Need: Downtime is minimized to minutes. The most recent application backup state is restored. (Depending on frequency of backup very little data may be lost)

Approach: FAR brings back the application instance on a stand-by or the original node near-instantly, minimizing down time. The application state at the time of backup is restored. Changes made after the last backup is are lost unless the application logs are available (either salvaged from the original node or from some replicated location). If current application logs are available and subsequently applied the application can be rolled forward to the time of failure with no loss of data.

PIT Image Used: Usually latest but depending on reason for disaster (for example: virus attack) an image preceding the event. Application Specific Notes:

Exchange 2003: Complicated scenarios like 'Dial Tone Recovery' involving creation of an empty database and then switching databases when recovery to the RSG (Recovery Storage Group) is done and then re-merging is no longer needed since FAR is quick and painless reducing application outage to a minimum. SQL Server: Stand-by databases, replication, and/or log-shipping are expensive and administration intensive options for SQL Server availability. FAR is an option that is easy to deploy with reduced administration cost combining the power of fast backups and quick availability on demand. Example 2a. With Online Restore Need: Application data needs to be finally restored back to local or SAN attached storage. Using storage from secondary storage may be only a temporary option.

Where Performed: Usually to the original application node or a proximate node depending on nature of disaster and preventive setup. Follow-Up (LAR): The application is online and users can start using the application within minutes. Restore continues in the background to a local disk slice while the application is up and running. After all data is restored to the local slice, the application is stopped or paused briefly and the iSCSI mappings are removed. Then

the local slice is promoted to be the sole application disk. The application is resumed or restarted. The application is unavailable (if at all) only for a brief period at the end.

BAR - Regular backup schedule for protecting the application on the newly restore volume kicks in. (The cycle repeats if the application needs to be restored in the future)

Example 2b. Without Online Restore.

Need: The reason that no background restore is needed is either that the standby node is temporary and degraded performance is adequate (fail-back may be in the offing once the original site has been re-constructed) or that the filer storing the backup image is powerful enough to host the application.

Redundant destination: A high end filer (possibly at a remote site) can mirror the backup images stored on the original backup destination (for example, to tertiary storage). This configuration lends itself to restore being redirected to the high-end filer and not the original filer. Background restore to a local slice is not needed in this case as the filer storage would be high-end and permanent. Quality of restored storage:

A. Low - iSCSI mounts to secondary storage: Applications may be able to survive moderately performing storage over iSCSI, especially if this is a temporary situation and Fail Back is anticipated shortly once higher quality storage and nodes are repaired or independently restored.

B. High - iSCSI mount to high performance storage established by Secondary to Tertiary Replication, or copy to from original filer following backup: Applications will perform adequately and this may be a permanent solution. This does not preclude failback however. Follow-Up: If needed the backup after restore (BAR) could continue from the target machine or a NAS block-level backup may be initiated since the storage has been effectively migrated to the NAS device. The LUNs on the filer may be cloned to break their dependency from the original snapshots since permanent storage on the filer has been established with its own storage life-cycle. ERF (Eventual Rapid Failback) for APPs :

Applications may eventually fail back to the original node or to a separate recreated node in the following manner:

1. Shutdown Application briefly on currently running node.

2. If a relationship was established between secondary and alternate storage and the original secondary is in the proximity of the final destination reverse the replication source and destination, resync, and update secondary from current storage. Else go to step 3. (This process works off the latest common snapshot and copies changes since then. This should complete quickly assuming fail back was initiated reasonably soon after the point of failure)

3. Perform FAR to desired node.

4. Application instance would be back to the state (with the latest changes) that it was on the stand-by node and normal operations could resume.

Example 3. Fine Grain Restore from Whole Application Backup

Need: For most applications fine grain restores are not possible from a backup of the entire application. Granular application object backups are unreliable and extremely resource intensive. Given the state of the art of current backup/restore solutions of fine-grain application objects performing a FAR for the application to an alternate instance (which completes very quickly) and then using application specific tools to recover fine-grain objects is an extremely attractive option.

Approach: FAR followed by application specific tools to drill down and examine application objects. These can then be merged into the original destination or extracted for external use.

PIT Image Used: Depends on when a fine grain object was deleted or was in an uncorrupted state.

Where Performed: Usually to an alternate instance on a different node or to the original node (depending on setup and need). Folio w-Up: Usually nothing as the need is temporary and the instance is torn down and iSCSI mappings undone. Application Specific Notes: (

Exchange 2000 : Single mailbox restore without paying any backup penalties is possible using FAR and then using EXMERGE.EXE or other tools. Exchange 2003: The powerful combination of Recovery Storage Group and

FAR make single mailbox or even sub-mail box restore for fine-grain restore from any point in the past an extremely quick and painless option.

SQL Server: Table level restore - 'bcp' or other tools may be used to restore tables from an alternate FARed instance.

Example 4. Instant Replica for Apps for Analysis, Reporting, and Data Warehousing

Need: Typically obtaining a second copy of data for analysis or reporting is a luxury afforded large businesses who have implemented expensive split mirror technology with plenty of disk space. With FAR not only is this feasible at a much lowered cost but can be done near instantly to multiple destinations. Businesses would be empowered to explore more analytical possibilities and gain a competitive edge.

Approach: Using FAR to one or more nodes as frequently as desired. PIT Image Used: Usually latest but depending on analytical or business reasons some point in time in the past (perhaps data for last Christmas Sales)

Where Performed: To an alternate node. The original node still continues to run the line of business application.

What happens next (LAR)?: If the replica needs to have its own timeline or longevity it needs to be backed up. Backup continues with incremental changes from the restored copy. Example 5. Alternate Node restore for Tape Backup for Long Term Retention

Need: Additional protection and/or long term retention may require tape backup. Nearline images expire quickly thus tape backups are almost always necessary for long term retention.

Approach: Image backup of iSCSI mapped volumes to tape. The tape image can then restored at any granularity to any node at point in time in the future.

PIT Image Used: Usually staggered from the backup schedule and dictated by how many instances need to remain nearline. Where Performed: Some tape connected to stand-by node. This could also be an IV for APPs node.

Follow-Up: Image backup to tape is performed of the FAR volume(s) (License needed). After successful backup iSCSI mappings are removed and the stage is set for the next cycle. Example 6. FAR for Storage Migration

Need: There may be need to migrate direct attached or legacy SAN storage to block-oriented NAS Filer storage for cost, consolidation, performance, or manageability reasons.

Approach: Once a block-level backup has been done to the filer — the migration has already been seeded. The backup images may be copied or snap- mirrored to a high-end filer to further ease the process. FAR effectively completes the migration process. PIT Image Used: Usually latest.

Where Performed: To the new application node which will attach to the LUNs created on the filer.

Follow-Up: The LUNs will then be cloned (in the background) while the application is up and running to free them from the bondage of the snapshot containing them. The snapshots can then be re-cycled to reclaim space. Backup after restore (BAR) can then resume of the volumes backed by the LUN or of filer volumes or quota-trees containing the LUN. Example 7. FAR4C - FAR for Compliance

Need: Legal reasons. Typically compliance involves expensive solutions involving proprietary hardware. Backup Express image backup to secondary WORM storage provides an affordable solution which can recreate a machine state sometime in the past instantly and accurately.

Approach: FAR to stand-by node either recreating application state or entire machine state. PIT Image Used: Depends on whether this is needed for annual reports or on demand (which may be any point in time in the past depending on reason for scrutiny

Where Performed: Any stand-by node.

Follow-Up: Usually transient and torn down after regulators have been satisfied. The whole machine state can be archived to WORM tapes if needed via Scenario 5 for offline examination or portable compliance. A Further Example

Fig- 4(A & B) illustrates an instant availability and recovery scenario that utilizes Instant Availability to virtually eliminate business interruption during the recovery process:

• 11:00 a.m. Shows the last routine backup on the NAS 107 before disk failure on the primary node 300. .

• 12:00 p.m. Volume D 406 fails. 12:05 p.m. Within minutes, the 11:00 a.m. backup instance, accessed through a Logical Unit Number (LUN) 411 on the secondary storage unit is mapped via iSCSI (412) to drive letter D. Business

continues. The iSCSI connection to the secondary storage unit 107 is transparent to users. Note that data changes are stored in a "live data area" 414 on the secondary storage unit (square with white background blocks). The 11:00 a.m. backup instance itself 413 is read-only and does not change. • 12:05-1:00 p.m. The failed disk 406 is replaced with new disk 421. Normal business use continues via the live iSCSI connection to the secondary storage unit 107.

• 1 :00- 1 :45 p.m. The 11 :00 a.m. backup instance is transferred (451) to the primary 300 and its new disk, 421. Business continues via the live iSCSI connection without interruption until the system is brought down at 2:45 a.m.

• 2:45-3:00 a.m. Administrator performs data ^synchronization ("Lazy Mirror") (452). During this period, the system is unavailable to users. Instant Availability gives administrators the flexibility to perform the resynching (452) during an overnight maintenance period. • 3 :00 a.m. Recovery is completed. The Instant Availability connection is ended by remapping volume D to the new disk 421.

It is evident that the embodiments described herein accomplish the stated objects of the invention. While the presently preferred embodiments have been described in detail, it will be apparent to those skilled in the art that the principles of the invention are realizable by other devices, systems and methods without departing from the scope and spirit of the invention, as be defined in the following claims.

Title:

Method and system for data recovery in a continuous data protection system

Document Type and Number:
United States Patent 7325159
Link to this page:
http://www.freepatentsonline.com/7325159.html

Abstract:

In a continuous data protection system having a primary volume and a secondary volume, a method for data recovery begins by selecting a snapshot of the primary volume to be recovered and a location on which the snapshot is to be loaded. A point in time (PIT) map is created for the selected snapshot, and the selected snapshot is loaded at the selected location. A data block from the PIT map is resolved to determine which block on the secondary volume is presented as being part of the snapshot. The snapshot is accessed via a host computer as if the snapshot was the primary volume at an earlier point in time, corresponding to the time of the selected snapshot.

Inventors:

Stager, Roger Keith (Livermore, CA, US)
Trimmer, Donald Alvin (Livermore, CA, US)
Saxena, Pawan (Pleasanton, CA, US)
Johnson, Randall (Pleasant Grove, UT, US)
Johnston, Craig Anthony (Livermore, CA, US)
Chang, Yafen Peggy (Fremont, CA, US)
Blaser, Rico (San Francisco, CA, US)

Application Number:

10/772017

Filing Date:

02/04/2004

Publication Date:

01/29/2008

Assignee:

Network Appliance, Inc. (Sunnyvale, CA, US)

Primary Class:

714/2

Other Classes:

714/13

International Classes:

G06F11/00

Field of Search:

714/12, 707/202, 714/8, 714/11, 714/2, 714/13, 714/5

US Patent References:

4635145	January, 1987	Horie et al.		Floppy disk drive with stand-by mode
4727512	February, 1988	Birkner et al.		Interface adaptor emulating magnetic tape drive
4775969	October, 1988	Osterlund		Optical disk storage format, method and apparatus for emulating a magnetic tape drive
5235695	August, 1993	Pence		Apparatus for efficient utilization of removable data recording media
5297124	March, 1994	Plotkin et al.		Tape drive emulation system for a disk drive
5438674	August, 1995	Keele et al.		Optical disk system emulating magnetic tape units
5455926	October, 1995	Keele et al.		Virtual addressing of optical storage media as magnetic tape equivalents
5485321	January, 1996	Leonhardt et al.		Format and method for recording optimization
5666538	September, 1997	DeNicola		Disk power manager for network servers
5673382	September, 1997	Cannon et al.	714/6	Automated management of off-site storage volumes for disaster recovery
5774292	June, 1998	Georgiou et al.		Disk drive power management system and method
5774715	June, 1998	Madany et al.		File system level compression using holes
5805864	September, 1998	Carlson et al.		Virtual integrated cartridge loader for virtual tape storage system
5809511	September, 1998	Peake		Outboard data migration in a volume stacking library
5809543	September, 1998	Byers et al.		Fault tolerant extended processing complex for redundant nonvolatile file caching
5854720	December, 1998	Shrinkle et al.		Low-power hard disk drive system architecture

5864346	January, 1999	Yokoi et al.	Picture display unit and image display system
5872669	February, 1999	Morehouse et al.	Disk drive apparatus with power conservation capability
5875479	February, 1999	Blount et al.	Method and means for making a dual volume level copy in a DASD storage subsystem subject to updating during the copy interval
5911779	June, 1999	Stallmo et al.	Storage device array architecture with copyback cache
5949970	September, 1999	Sipple et al.	Dual XPCS for disaster recovery
5961613	October, 1999	DeNicola	Disk power manager for network servers
5963971	October, 1999	Fosler et al.	Method and apparatus for handling audit requests of logical volumes in a virtual media server
6021408	February, 2000	Ledain et al.	Methods for operating a log device
6023709	February, 2000	Anglin et al.	Automated file error classification and correction in a hierarchical storage management system
6029179	February, 2000	Kishi	Automated read-only volume processing in a virtual tape server
6041329	March, 2000	Kishi	Automated message processing system configured to automatically manage introduction of removable data storage media into media library
6044442	March, 2000	Jesionowski	External partitioning of an automated data storage library into multiple virtual libraries for access by a plurality of hosts
6049848	April, 2000	Yates et al.	System and method for performing high-speed tape positioning operations
6061309	May, 2000	Gallo et al.	Method and apparatus for maintaining states of an operator panel and convenience input/output station of a dual library manager/dual accessor controller system in the event of a failure to one controller

6067587	May, 2000	Miller et al.		Method for serializing and synchronizing data packets by utilizing a physical lock system and a control data structure for mutual exclusion lock
6070224	May, 2000	LeCrone et al.		Virtual tape system
6098148	August, 2000	Carlson		Storage and access of data using volume trailer
6128698	October, 2000	Georgis		Tape drive emulator for removable disk drive
6131142	October, 2000	Kamo et al.		Disk system and power-on sequence for the same
6131148	October, 2000	West et al.		Snapshot copy of a secondary volume of a PPRC pair
6163856	December, 2000	Dion et al.		Method and apparatus for file system disaster recovery
6163858	December, 2000	Dion et al.		Diagnostic methodology for debugging integrated software
6173359	January, 2001	Carlson et al.		Storage and access to scratch mounts in VTS system
6195730	February, 2001	West		Computer system with storage device mapping input/output processor
6225709	May, 2001	Nakajima		Power supply circuit for electric device
6247096	June, 2001	Fisher et al.		Handling eject requests of logical volumes in a data storage subsystem
6260110	July, 2001	LeCrone et al.		Virtual tape system with variable size
6266784	July, 2001	Hsiao et al.	714/6	Direct storage of recovery plan file on remote server for disaster recovery and storage management thereof
6269423	July, 2001	Kishi		Method and apparatus for providing improved caching for a virtual tape server
6269431	July, 2001	Dunham		Virtual storage and block level direct access of secondary storage for recovery of backup data
6282609	August, 2001	Carlson		Storage and access to scratch mounts in VTS system
6289425	September, 2001	Blendermann et al.		Method for verifying availability of data space in virtual tape system
6292889	September, 2001	Fitzgerald et al.		Distributed computer network including hierarchical resource information structure and related method of distributing

			resources
6301677	October, 2001	Squibb	System and apparatus for merging a write event journal and an original storage to produce an updated storage using an event map
6304880	October, 2001	Kishi	Automated reclamation scheduling override in a virtual tape server
6317814	November, 2001	Blendermann et al.	Method for selectively storing redundant copies of virtual volume data on physical data storage cartridges
6324497	November, 2001	Yates et al.	Tape drive emulation system including tape library interface
6327418	December, 2001	Barton	Method and apparatus implementing random access and time-based functions on a continuous stream of formatted digital data
6336163	January, 2002	Brewer et al.	Method and article of manufacture for inserting volumes for import into a virtual tape server
6336173	January, 2002	Day et al.	Storing and tracking multiple copies of data in data storage libraries
6339778	January, 2002	Kishi	Method and article for apparatus for performing automated reconcile control in a virtual tape system
6341329	January, 2002	LeCrone et al.	Virtual tape system
6343342	January, 2002	Carlson	Storage and access of data using volume trailer
6353837	March, 2002	Blumenau	Method and apparatus providing mass storage access from systems using different meta-data formats
6360232	March, 2002	Brewer et al.	Disaster recovery method for a removable media library
6389503	May, 2002	Georgis et al.	Tape drive emulation by removable disk drive and media formatted therefor
6408359	June, 2002	Ito et al.	Storage device management system and method for distributively storing data in a plurality of storage devices
6487561	November,	Ofek et al.	Apparatus and methods for copying, backing up, and restoring data using a backup

	2002			segment size larger than the storage block size
6496791	December, 2002	Yates et al.		Interfaces for an open systems server providing tape drive emulation
6499026	December, 2002	Rivette et al.		Using hyperbolic trees to visualize data generated by patent-centric and group-oriented data processing
6557073	April, 2003	Fujiwara		Storage apparatus having a virtual storage area
6557089	April, 2003	Reed et al.		Backup by ID-suppressed instant virtual copy then physical backup copy with ID reintroduced
6578120	June, 2003	Crockett et al.		Synchronization and resynchronization of loosely-coupled copy operations between a primary and a remote secondary DASD volume under concurrent updating
6615365	September, 2003	Jenevein et al.		Storing a computer disk image within an imaged partition
6625704	September, 2003	Winokur		Data backup method and system using snapshot and virtual tape
6654912	November, 2003	Viswanathan et al.		Recovery of file system data in file servers mirrored file system volumes
6658435	December, 2003	McCall		Disk image backup/restore with data preparation phase
6694447	February, 2004	Leach et al.	714/6	Apparatus and method for increasing application availability during a disaster fail-back
6725331	April, 2004	Kedem		Method and apparatus for managing the dynamic assignment resources in a data storage system
6766520	July, 2004	Rieschl et al.		Tape drive emulation software objects, and emulation of other peripheral systems for computers
6779057	August, 2004	Masters et al.		Method, system, and program for indicating data transmitted to an input/output device as committed
6779058	August, 2004	Kishi et al.		Method, system, and program for transferring data between storage devices

6779081	August, 2004	Arakawa et al.		Apparatus and method for defragmentation in disk storage system
6816941	November, 2004	Carlson et al.		Method and system for efficiently importing/exporting removable storage volumes between virtual storage systems
6816942	November, 2004	Okada et al.		Storage control apparatus and method for compressing data for disk storage
6834324	December, 2004	Wood		System and method for virtual tape volumes
6850964	February, 2005	Brough et al.		Methods for increasing cache capacity utilizing delta data
6915397	July, 2005	Lubbers et al.		System and method for generating point in time storage copy
6931557	August, 2005	Togawa		Information processing apparatus, power control method and recording medium to control a plurality of driving units according to the type of data to be processed
6950263	September, 2005	Suzuki et al.		Storage apparatus and control method therefor
6973534	December, 2005	Dawson		Apparatus and method to export and then import a logical volume with assigned storage attributes
6978325	December, 2005	Gibble		Transferring data in virtual tape server, involves determining availability of small chain of data, if large chain is not available while transferring data to physical volumes in peak mode
7032126	April, 2006	Zalewski et al.	714/7	Method and apparatus for creating a storage pool by dynamically mapping replication schema to provisioned storage volumes
7055009	May, 2006	Factor et al.		Method, system, and program for establishing and maintaining a point-in-time copy
7096331	August, 2006	Haase et al.		System and method for managing data associated with copying and replication procedures in a data storage environment

7100089	August, 2006	Phelps		Determining differences between snapshots
7111136	September, 2006	Yamagami		Method and apparatus for backup and recovery system using storage based journaling
7127388	October, 2006	Yates et al.		Interfaces for an open systems server providing tape drive emulation
7155586	December, 2006	Wagner et al.	711/162	Method of allowing point-in-time view of data on a disk using a map on cache disk
20020004835	January, 2002	Yarbrough		Message queue server system
20020016827	February, 2002	McCabe et al.		Flexible remote data mirroring
20020026595	February, 2002	Saitou et al.		Power supply control system and power supply control method capable of reducing electric power consumption
20020095557	July, 2002	Constable et al.		Virtual data storage (VDS) system
20020144057	October, 2002	Li et al.		Archival data storage system and method
20020166079	November, 2002	Ulrich et al.		Dynamic data recovery
20020199129	December, 2002	Bohrer et al.		Data storage on a computer disk array
20030004980	January, 2003	Kishi et al.		Preferential caching of uncopied logical volumes in a peer-to-peer virtual tape server
20030037211	February, 2003	Winokur		Data backup method and system using snapshot and virtual tape
20030120676	June, 2003	Holavanahalli et al.		Methods and apparatus for pass-through data block movement with virtual storage appliances
20030126388	July, 2003	Yamagami		Method and apparatus for managing storage based replication
20030135672	July, 2003	Yip et al.		System having tape drive emulator and data cartridge carrying a non-tape storage medium
20030149700	August, 2003	Bolt		Emulated backup tape drive using data compression
20030182350	September, 2003	Dewey		Method,system, and program for allocating tasks to a plurality of processors
				System, method, and

				architecture for dynamic server power management and dynamic workload management for multi-server environment
20030188208	October, 2003	Fung		
20030225800	December, 2003	Kavuri		Selective data replication system and method
20040015731	January, 2004	Chu et al.		Intelligent data management fo hard disk drive
20040098244	May, 2004	Dailey et al.		Method and system for emulating tape storage format using a non-tape storage medium
20040181388	September, 2004	Yip et al.		System having tape drive emulator and data tape cartridge housing carrying multiple disk drives
20040181707	September, 2004	Fujibayashi	714/6	Method and apparatus for seamless management for disaster recovery
20050010529	January, 2005	Zalewski et al.	705/54	Method and apparatus for building a complete data protection scheme
20050063374	March, 2005	Rowan et al.		Method for identifying the time at which data was written to a data store
20050065962	March, 2005	Rowan et al.		Virtual data store creation and use
20050066118	March, 2005	Perry et al.		Methods and apparatus for recording write requests directed to a data store
20050066222	March, 2005	Rowan et al.		Systems and methods for time dependent data storage and recovery
20050066225	March, 2005	Rowan et al.		Data storage system
20050076264	March, 2005	Rowan et al.		Methods and devices for restoring a portion of a data store
20050076070	April, 2005	Mikami		Method, apparatus, and computer readable medium for managing replication of back-up object
20050076261	April, 2005	Rowan et al.		Method and system for obtaining data stored in a data store
20050076262	April, 2005	Rowan et al.		Storage management device
20050144407	June, 2005	Colgrove et al.		Coordinated storage management operations in replication environment

20060047895	March, 2006	Rowan et al.	Systems and methods for providing a modification history for a location within a data store
20060047902	March, 2006	Passerini	Processing storage-related I/O requests using binary tree data structures
20060047903	March, 2006	Passerini	Systems, apparatus, and methods for processing I/O requests
20060047905	March, 2006	Matze et al.	Tape emulating disk based storage system and method with automatically resized emulated tape capacity
20060047925	March, 2006	Passerini	Recovering from storage transaction failures using checkpoints
20060047989	March, 2006	Delgado et al.	Systems and methods for synchronizing the internal clocks of a plurality of processor modules
20060047998	March, 2006	Darcy	Methods and apparatus for optimally selecting a storage buffer for the storage of data
20060047999	March, 2006	Passerini et al.	Generation and use of a time map for accessing a prior image of a storage device
20060143376	June, 2006	Matze et al.	Tape emulating disk based storage system and method

Foreign References:

EP1333379	April, 2006	Emulated backup tape drive using data compression
EP1671231	June, 2006	SYSTEMS AND METHODS FOR TIME DEPENDENT DATA STORAGE AND RECOVERY
EP1671231	June, 2006	SYSTEMS AND METHODS FOR TIME DEPENDENT DATA STORAGE AND RECOVERY
WO/1999/003098	January, 1999	IMPROVED INTERFACES FOR AN OPEN SYSTEMS SERVER PROVIDING TAPE DRIVE EMULATION
WO/1999/006912	February, 1999	TAPE DRIVE EMULATION BY REMOVABLE DISK DRIVE AND MEDIA FORMATTED THEREFOR
WO/2005/031576	April, 2005	SYSTEMS AND METHODS FOR TIME DEPENDENT DATA STORAGE AND RECOVERY
WO/2006/023990	March, 2006	RECOVERING FROM STORAGE TRANSACTION FAILURES USING CHECKPOINTS

WO/2006/023991	March, 2006	SYSTEMS AND METHODS FOR PROVIDING A MODIFICATION HISTORY FOR A LOCATION WITHIN A DATA STORE
WO/2006/023992	March, 2006	IMAGE DATA STROAGE DEVICE WRITE TIME MAPPING
WO/2006/023993	March, 2006	DATA STORAGE SYSTEM
WO/2006/023994	March, 2006	METHODS AND DEVICES FOR RESTORING A PORTION OF A DATA STORE
WO/2006/023995	March, 2006	METHODS AND APPARATUS FOR RECORDING WRITE REQUESTS DIRECTED TO A DATA STORE

Other References:

"Alacritus Software's Securitus I: Pointing the Way to Virtual Tape Libraries" Aberdeen Group, Inc., Mar. 2002.
"Continuous Data Protection: Business Continuity for the Era of Networked Storage: An Executive White Paper" Aberdeen Group, Inc., Jul. 2003.
"Alacritus Software's Chronospan: Make Time for Continuous Data Protection" Aberdeen Group, Inc., Oct. 2003.
Hill, David "Alacritus Software's Securitus: Defining the Way to Virtual Tape Libraries" Aberdeen Group, Inc., Jul. 2003.
"Alacritus Software's Securitus: Defining the Way to Virtual Tape Libraries" Aberdeen Group, Inc., Jul. 2003.
"Product Brief: Rhapsody/Alacritus-Secritus/XPath Virtual Tape in the Fabric" The Enterprise Storage Group, Aug. 2002.
"Alacritus Software Announces Securitus I, The Industry's First Virtual Tape Library Solution: Securitus I Heralds Advent of 'Disruptive Technology' that Serves as Replacement to Tape Libraries" Alacritus Software, Inc., Jun. 25, 2001.
"Alacritus, Hitachi CP and Nissho Team to Create Virtual Tape Library Appliance: Industry's First Virtual Tape Library Appliance to Replace Storage Tape Libraries" Alacritus Software, Inc., Jun. 25, 2001.
"Hitachi CP, Nissho, and Alacritus Software Bring Virtual Tape Library Appliance Solution to Market: Three Companies Join to Deliver VTLA Smart Guard—A Disk Subsystem Product that Functions as a Virtual Storage Tape Library" Alacritus Software, Inc., Oct. 3, 2001.
Trimmer, Don, "Tape Free Backup/Recovery: Requirements and Advantages: Virtualization Technology Will Encompass Many Applications, One of the Most Significant Possibly Being Backup/Recovery" InfoStor, Mar. 2002.
"Alacritus Software Announces Virtual Tape Library Support for Legato Networker Data Protection Solution" Alacritus Software, Inc., Jan. 8, 2002.
Camphuisen, Alicia, "Hitachi Inks OEM Deal with Legato" Knapp Comm., Jul. 17, 2002.
"Alacritus Announces Disk-Based Successor to Tape" Knapp Comm., Aug. 21, 2002.
Biggar, Heidi, "Alacritus Enables Disk-Based Backup" InfoStor, Sep. 2001.
"Securitus I White Paper: Disk Based Data Protection from Alacritus Software" Alacritus Software, Inc., Jul. 2001.
"Alacritus Software FAQs" Alacritus Software, Inc., Jul. 2001.
"Disk-Based Data Protection" Alacritus Software, Inc., Jul. 2001.
"Virtual Tape Library Technology Brochure" Alacritus Software, Inc., Jul. 2001.
"Disk-Based Data Protection" Alacritus Security, Inc., Sep. 2001.
"Disk-Based Data Protection" Alacritus Software, Inc., Sep. 2002.
Payack, Paul JJ, "Alacritus Lines Up OEM Partners for Virtual Tape Library

Push" The (451) Storage & Systems, Oct. 4, 2002.
Payack, Paul JJ, "Alacritus Software Announces Continuous Data Protection with New Chronospan Technology" Oct. 28, 2003.
Payack, Paul JJ, "Alacritus Software Announces New Customers for Securitus VTLA" Alacritus Software, Jan. 13, 2004.
Baltazar, Henry "Weaving Apps Into SAN Fabric" eWEEK, Mar. 24, 2003.
Baltazar, Henry "More Intelligence is on the Way" eWEEK, Sep. 15, 2003.
Barrett, Alex "The Case for Network Smarts" Storage Magazine, Jun. 2003.
"Securitus White Paper: Disk Based Data Protection from Alacritus Software" Alacritus Website, Oct. 2003.
"Manageability: Securitus v. Tape" Alacritus Website, Oct. 2003.
"The SNIA Data Management Forum Created to Tackle Data Protection and Information Lifecycle Management Issues: Enhanced Backup Solutions Initiative Rolls Efforts into New SNIA Forum" Storage Networking Industry Association, Oct. 13, 2003.
"No Changes Required: Securitus v. Tape" Alacritus Website, Oct. 2003.
"Customer Success" Alacritus Website, Oct. 2003.
"Chronospan" Alacritus Website, Oct. 2003.
"Alacritus Software Announces Securitus I, the Industry's First Virtual Tape Library Solution: Securitus I Heralds Advent of 'Disruptive Technology' that Serves as Replacement to Tape Libraries" Alacritus Software, Inc., Apr. 9, 2002.
Biggar, Heidi, "Disk and Tape Forge New Partnership in Backup Arena" InfoStor, Nov. 2001.
Preston, W. Curtis, "Surprise! Cheap Disks Cure Slow Backup" Storage Magazine, Jun. 1, 2002.
"Alacritus, Hitachi CP and Nissho Team to Create Virtual Tape Library" internetnews.com, Jun. 25, 2001.
"Alacritus Software and Rhapsody Networks to Develop Breakthrough Backup Solutions for Storage Networks: Companies to Provide First Network-Intelligent Virtual Tape Solution Resulting in Dramatic ROI, Increases in Data Backup Performance and Scalability" Alacritus Software, Jul. 8, 2002.
Korniega, Kevin, "Vendor Pushes Disk Backup Over Tape" SearchStorage.com Jan. 10, 2003.
"Testimonials" Alacritus Website, Oct. 2003.
"Seamless Integration" Alacritus Website, Oct. 2003.
"Topologies" Alacritus Website, Oct. 7, 2003.
"Securitus" Alacritus Website, Oct. 2003.
"Scalability: Securitus v. Tape" Alacritus Website, Oct. 2003.
"Strengths: Securitus v. Tape" Alacritus Website, Oct. 2003.
"Alacritus Software's Securitus I: Pointing the Way to Virtual Tape Libraries" Aberdeen Group, Inc., Mar. 2002.
"Continuous Data Protection: Business Continuity for the Era of Networked Storage: An Executive White Paper" Aberdeen Group, Inc., Jul. 2003.
"Alacritus Software's Chronospan: Make Time for Continuous Data Protection" Aberdeen Group, Inc., Oct. 2003.
Hill, David "Alacritus Software's Securitus: Defining the Way to Virtual Tape Libraries" Aberdeen Group, Inc., Jul. 2003.
"Alacritus Software's Securitus: Defining the Way to Virtual Tape Libraries" Aberdeen Group, Inc. Jul. 2003.
"Product Brief: Rhapsody/Alacritus-Secritus/XPath Virtual Tape in the Fabric" The Enterprise Storage Group, Aug. 2002.
"Alacritus Software Announces Securitus I, The Industry's First Virtual Tape Library Solution: Securitus I Heralds Advent of 'Disruptive Technology' that Serves as Replacement to Tape Libraries" Alacritus Software, Inc., Jun. 25, 2001.

"Alacritus, Hitachi CP and Nissho Team to Create Virtual Tape Library Appliance: Industry's First Virtual Tape Library Appliance to Replace Storage Tape Libraries" Alacritus Software, Inc., Jun. 25, 2001.
"Hitachi CP, Nissho, and Alacritus Software Bring Virtual Tape Library Appliance Solution to Market: Three Companies Join to Deliver VTLA Smart Guard - A Disk Subsystem Product that Functions as a Virtual Storage Tape Library" Alacritus Software, Inc., Oct. 3, 2001.
Trimmer, Don, "Tape Free Backup/Recovery: Requirements and Advantages: Virtualization Technology Will Encompass Many Applications, One of the Most Significant Possibly Being Backup/Recovery" InfoStor, Mar. 2002.
"Alacritus Software Announces Virtual Tape Library Support for Legato NetWorker Data Protection Solution" Alacritus Software, Inc., Jan. 8, 2002.
Camphuisen, Alicia, "Hitachi Inks OEM Deal with Legato" Knapp Comm., Jul. 17, 2002.
Biggar, Heidi, "Alacritus Enables Disk-Based Backup" InfoStor, Sep. 2001.
"Securitus I White Paper: Disk Based Data Protection from Alacritus Software" Alacritus Software, Inc., Jul. 2001.
"Alacritus Software FAQs" Alacritus Software, Inc., Jul. 2001.
"Disk-Based Data Protection" Alacritus Software, Inc., Jul. 2001.
"Virtual Tape Library Technology Brochure" Alactritus Software, Inc., Jul. 2001.
"Disk-Based Data Protection" Alacritus Software, Inc., Sep. 2001.
"Disk-Based Data Protection" Alacritus Software, Inc., Sep. 2002.
Payack, Paul JJ, "Alacritus Lines Up OEM Partners for Virtual Library Push" The (451) Storage & Systems, Oct. 4, 2002.
Payack, Paul JJ, "Alacritus Software Announces Continuous Data Protection with New Chronospan Technology" Oct. 28, 2003.
Payack, Paul JJ, "Alacritus Software Announces New Customers for Securitus VTLA" Alacritus Software, Jan. 13, 2004.
Baltazar, Henry "Weaving Apps Into SAN Fabric" eWEEK, Mar. 24, 2003.
Baltazar, Henry "More Intelligence is on the Way" eWEEK, Sep. 15, 2003.
Barrett, Alex "The Case for Network Smarts" Storage Magazine, Jun. 2003.
"Securitus White Paper: Disk Based Data Protection from Alacritus Software" Alacritus Website, Oct. 2003.
"Managability: Securitus v. Tape" Alacritus Website, Oct. 2003.
"The SNIA Data Management Forum Created to Tackle Data Protection and Information Lifecycle Management Issues: Enhanced Backup Solutions Initiative Rolls Efforts into New SNIA Forum" Storage Networking Industry Association, Oct. 13, 2003.
"No Changes Required: Securitus v. Tape" Alacritus Website, Oct. 2003.
"Customer Success" Alacritus Website, Oct. 2003.
"Alacritus Software Announces Securitus I, the Industry's First Virtual Tape Library Solution: Securitus I Heralds Advent of 'Disruptive Technology' that Serves as Replacement to Tape Libraries" Alacritus Software, Inc., Apr. 9, 2002.
Biggar, Heidi, "Disk and Tape Forge New Partnership in Backup Arena" InfoStor, Nov. 2001.
Preston, W. Curtis, "Surprise! Cheap Disks Cure Slow Backup" Storage Magazine, Jun. 1, 2002.
"Alacritus, Hitachi CP and Nissho Team to Create Virtual Tape Library" internetnews.com, Jun. 25, 2001.
"Alacritus Software and Rhapsody Networks to Develop Breakthrough Backup Solutions for Storage Networks: Companies to Provide First Network-Intelligent Virtual Tape Solution Resulting in Dramatic ROI, Increases in Data Backup Performance and Scalability" Alacritus Software, Jul. 8, 2002.
Komiega, Kevin, "Vendor Pushes Disk Backup Over Tape" SearchStorage.com

Jan. 10, 2003.
"Testimonials" Alacritus Website, Oct. 2003.
"Seamless Integration" Alacritus Website, Oct. 2003.
"Topologies" Alacritus Website, Oct. 7, 2003.
"Securitus" Alacritus Website, Oct. 2003.
"Scalability: Securitus v. Tape" Alacritus Website, Oct. 2003.
"Strengths: Securitus v. Tape" Alacritus Website, Oct. 2003.
"Alacritus Announces Disk-Based Successor to Tape" Knapp Comm., Aug. 21, 2002.

Primary Examiner:

Le, Dieu-minh

Attorney, Agent or Firm:

Volpe and Koenig, P.C.

Parent Case Data:

CROSS REFERENCE TO RELATED APPLICATION(S)

This application claims priority from U.S. Provisional Application No. 60/541,626, entitled "METHOD AND SYSTEM FOR CONTINUOUS DATA PROTECTION," filed on Feb. 4, 2004, which is incorporated by reference as if fully set forth herein.

Claims:

What is claimed is:

1. A method for data recovery in a continuous data protection system, comprising the steps of: selecting a snapshot of a primary volume in the continuous data protection system, the snapshot indicating the data on the primary volume at an earlier point in time which is to be recovered; choosing a location on which the snapshot is to be loaded; creating a point in time (PIT) map corresponding to the selected snapshot; and loading the selected snapshot at the chosen location, thereby making the data which was on the primary volume at a previous point in time accessible at the chosen location.

2. The method according to claim 1, wherein the selecting step includes selecting a scheduled snapshot.

3. The method according to claim 1, wherein the selecting step includes selecting an any point in time (APIT) snapshot.

4. The method according to claim 3, further comprising the step of creating a delta map spanning a time between the time of the selected APIT snapshot and a time of a second delta map, the second delta map immediately preceding the selected APIT snapshot, the delta map being created prior to creating the PIT map.

5. The method according to claim 4, wherein the created delta map includes changes made between the time of the selected APIT snapshot and the time of the second delta map.

6. The method according to claim 4, wherein the creating step includes merging the created delta map with all delta maps earlier than the second delta map.

7. The method according to claim 6, wherein the merging step is optimized by using pre-merged delta maps.

8. The method according to claim 1, wherein the choosing step includes choosing a logical unit on which the snapshot is to be loaded.

9. The method according to claim 1, further comprising the step of: accessing the snapshot via a host computer as if the snapshot was the primary volume at an earlier point in time, corresponding to the time of the selected snapshot.

10. The method according to claim 1, further comprising the step of: controlling access to the loaded snapshot, wherein only authorized host computers can access the loaded snapshot.

11. A system for data recovery in a continuous data protection system, comprising: a host computer; a primary data volume; and a continuous data protection system, configured to: select a snapshot of said primary data volume, said snapshot indicating the data on said primary volume at an earlier point in time which is to be recovered; choose a logical unit for loading said selected snapshot; load said selected snapshot at the chosen logical unit, thereby making the data which was on said primary volume at a previous point in time accessible at the chosen logical unit; and access said selected snapshot on the chosen logical unit via said host computer.

12. The system according to claim 11, wherein said continuous data protection system is configured to select a scheduled snapshot.

13. The system according to claim 11, wherein said continuous data protection system is configured to select an any point in time (APIT) snapshot.

14. The system according to claim 13, wherein said continuous data protection system is further configured to create a delta map spanning a time between the time of said selected APIT snapshot and a time of a second delta map, said second delta map immediately preceding said selected APIT snapshot.

15. The system according to claim 14, wherein said created delta map includes changes made between the time of said selected APIT snapshot and the time of said second delta map.

16. The system according to claim 11, wherein said continuous data protection system is further configured to control access to said selected snapshot.

17. The system according to claim 16, wherein said continuous data protection system is configured to permit only authorized host computers to access said selected snapshot.

18. A computer-readable storage medium containing a set of instructions for a general purpose computer, the set of instructions comprising: a selecting code segment for selecting a snapshot of a primary volume in a continuous data protection system, the snapshot indicating data on the primary volume at an earlier point in time which is to be recovered; a choosing code segment for choosing a location on which the snapshot is to be loaded; a creating code segment for creating a point in time (PIT) map corresponding to the selected snapshot; and a loading code segment for loading the selected snapshot at the selected location,

thereby making the data which was on the primary volume at a previous point in time accessible at the chosen location.

19. The storage medium according to claim 18, wherein the set of instructions further comprises: a second creating code segment for creating a delta map spanning a time between the time of a selected any point in time (APIT) snapshot and a time of a second delta map, the second delta map immediately preceding the selected APIT snapshot.

20. The storage medium according to claim 18, wherein the set of instructions further comprises: a controlling code segment for controlling access to the loaded snapshot, wherein only authorized host computers can access the loaded snapshot.

Description:

FIELD OF INVENTION

The present invention relates generally to continuous data protection, and more particularly, to data recovery in a continuous data protection system.

BACKGROUND

Hardware redundancy schemes have traditionally been used in enterprise environments to protect against component failures. Redundant arrays of independent disks (RAID) have been implemented successfully to assure continued access to data even in the event of one or more media failures (depending on the RAID Level). Unfortunately, hardware redundancy schemes are ineffective in dealing with logical data loss or corruption. For example, an accidental file deletion or virus infection is automatically replicated to all of the redundant hardware components and can neither be prevented nor recovered from by such technologies. To overcome this problem, backup technologies have traditionally been deployed to retain multiple versions of a production system over time. This allowed administrators to restore previous versions of data and to recover from data corruption.

Backup copies are generally policy-based, are tied to a periodic schedule, and reflect the state of a primary volume (i.e., a protected volume) at the particular point in time that is captured. Because backups are not made on a continuous basis, there will be some data loss during the restoration, resulting from a gap between the time when the backup was performed and the restore point that is required. This gap can be significant in typical environments where backups are only performed once per day. In a mission-critical setting, such a data loss can be catastrophic. Beyond the potential data loss, restoring a primary volume from a backup system can be complicated and often takes many hours to complete. This additional downtime further exacerbates the problems associated with a logical data loss.

The traditional process of backing up data to tape media is time driven and time dependent. That is, a backup process typically is run at regular intervals and covers a certain period of time. For example, a full system backup may be run once a week on a weekend, and incremental backups may be run every weekday during an overnight backup window that starts after the close of business and ends before

the next business day. These individual backups are then saved for a predetermined period of time, according to a retention policy. In order to conserve tape media and storage space, older backups are gradually faded out and replaced by newer backups. Further to the above example, after a full weekly backup is completed, the daily incremental backups for the preceding week may be discarded, and each weekly backup may be maintained for a few months, to be replaced by monthly backups. The daily backups are typically not all discarded on the same day. Instead, the Monday backup set is overwritten on Monday, the Tuesday backup set is overwritten on Tuesday, and so on. This ensures that a backup set is available that is within eight business hours of any corruption that may have occurred in the past week.

Despite frequent hardware failures and the necessity of ongoing maintenance and tuning, the backup creation process can be automated, while restoring data from a backup remains a manual and time-critical process. First, the appropriate backup tapes need to be located, including the latest full backup and any incremental backups made since the last full backup. In the event that only a partial restoration is required, locating the appropriate backup tape can take just as long. Once the backup tapes are located, they must be restored to the primary volume. Even under the best of circumstances, this type of backup and restore process cannot guarantee high availability of data..

Another type of data protection involves making point in time (PIT) copies of data. A first type of PIT copy is a hardware-based PIT copy, which is a mirror of the primary volume onto a secondary volume. The main drawbacks to a hardware-based PIT copy are that the data ages quickly and that each copy takes up as much disk space as the primary volume. A software-based PIT, typically called a "snapshot," is a "picture" of a volume at the block level or a file system at the operating system level. Various types of software-based PITs exist, and most are tied to a particular platform, operating system, or file system. These snapshots also have drawbacks, including occupying additional space on the primary volume, rapid aging, and possible dependencies on data stored on the primary volume wherein data corruption on the primary volume leads to corruption of the snapshot. In addition, snapshot systems generally do not offer the flexibility in scheduling and expiring snapshots that backup software provides.

While both hardware-based and software-based PIT techniques reduce the dependency on the backup window, they still require the traditional tape-based backup and restore process to move data from disk to tape media and to manage the different versions of data. This dependency on legacy backup applications and processes is a significant drawback of these technologies. Furthermore, like traditional tape-based backup and restore processes, PIT copies are made at discrete moments in time, thereby limiting any restores that are performed to the points in time at which PIT copies have been made.

A need therefore exists for a system that combines the advantages of tape-based systems with the advantages of snapshot systems and eliminates the limitations described above.

SUMMARY

In a continuous data protection system having a primary volume and a secondary volume, a method for data recovery begins by selecting a snapshot of the primary

volume to be recovered and a location on which the snapshot is to be loaded. A point in time (PIT) map is created for the selected snapshot, and the selected snapshot is loaded at the selected location. A data block from the PIT map is resolved to determine which block on the secondary volume is presented as being part of the snapshot. The snapshot is accessed via a host computer as if the snapshot was the primary volume at an earlier point in time, corresponding to the time of the selected snapshot.

A system for data recovery in a continuous data protection system includes a host computer, a primary data volume, and a secondary data volume. Creating means are used to create a snapshot of the primary data volume and storing means are used to store the snapshot on the secondary data volume. Selecting means are provided for selecting a snapshot on the secondary data volume. Accessing means are used to access the selected snapshot on the host computer, wherein the selected snapshot is presented to a user as if the selected snapshot were the primary data volume at an earlier point in time, corresponding to the time of the selected snapshot.

BRIEF DESCRIPTION OF THE DRAWING(S)

A more detailed understanding of the invention may be had from the following description of a preferred embodiment, given by way of example, and to be understood in conjunction with the accompanying drawings, wherein:

FIGS. 1A-1C are block diagrams showing a continuous data protection environment in accordance with the present invention;

FIG. 2 is an example of a delta map in accordance with the present invention;

FIG. 3 is a flowchart showing a data recovery procedure in accordance with the present invention; and

FIG. 4 is a diagram illustrating a retention policy for the fading out of snapshots in accordance with the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

In the present invention, data is backed up continuously, allowing system administrators to pause, rewind, and replay live enterprise data streams. This moves the traditional backup methodologies into a continuous background process in which policies automatically manage the lifecycle of many generations of restore images.

System Construction

FIG. 1A shows a preferred embodiment of a protected computer system **100** constructed in accordance with the present invention. A host computer **102** is connected directly to a primary data volume **104** (the primary data volume may

also be referred to as the protected volume) and to a data protection system **106** . The data protection system **106** manages a secondary data volume **108** . The construction of the system **100** minimizes the lag time by writing directly to the primary data volume **104** and permits the data protection system **106** to focus exclusively on managing the secondary data volume **108** . The management of the volumes is preferably performed using a volume manager.

A volume manager is a software module that runs on a server or intelligent storage switch to manage storage resources. Typical volume managers have the ability to aggregate blocks from multiple different physical disks into one or more virtual volumes. Applications are not aware that they are actually writing to segments of many different disks because they are presented with one large, contiguous volume. In addition to block aggregation, volume managers usually also offer software RAID functionality. For example, they are able to split the segments of the different volumes into two groups, where one group is a mirror of the other group. This is, in a preferred embodiment, the feature that the data protection system is taking advantage of when the present invention is implemented as shown in FIG. 1A. In many environments, the volume manager or host-based driver already mirrors the writes to two distinct different primary volumes for redundancy in case of a hardware failure. The present invention is configured as a tertiary mirror target in this scenario, such that the volume manager or host-based driver also sends copies of all writes to the data protection system.

It is noted that the primary data volume **104** and the secondary data volume **108** can be any type of data storage, including, but not limited to, a single disk, a disk array (such as a RAID), or a storage area network (SAN). The main difference between the primary data volume **104** and the secondary data volume **108** lies in the structure of the data stored at each location, as will be explained in detail below. It is noted that there may also be differences in terms of the technologies that are used. The primary volume **104** is typically an expensive, fast, and highly available storage subsystem, whereas the secondary volume **108** is typically cost-effective, high capacity, and comparatively slow (for example, ATA/SATA disks). Normally, the slower secondary volume cannot be used as a synchronous mirror to the high-performance primary volume, because the slower response time will have an adverse impact on the overall system performance.

The data protection system **106** , however, is optimized to keep up with high-performance primary volumes. These optimizations are described in more detail below, but at a high level, random writes to the primary volume **104** are processed sequentially on the secondary volume **108** . Sequential writes improve both the cache behavior and the actual volume performance of the secondary volume **108** . In addition, it is possible to aggregate multiple sequential writes on the secondary volume **108** , whereas this is not possible with the random writes to the primary volume **104** . The present invention does not require writes to the data protection system **106** to be synchronous. However, even in the case of an asynchronous mirror, minimizing latencies is important.

FIG. 1B shows an alternate embodiment of a protected computer system **120** constructed in accordance with the present invention. The host computer **102** is directly connected to the data protection system **106** , which manages both the primary data volume **104** and the secondary data volume **108** . The system **120** is likely slower than the system **100** described above, because the data protection system **106** must manage both the primary data volume **104** and the secondary data volume **108** . This results in a higher latency for writes to the primary volume

104 in the system **120** and lowers the available bandwidth for use. Additionally, the introduction of a new component into the primary data path is undesirable because of reliability concerns.

FIG. 1C shows another alternate embodiment of a protected computer system **140** constructed in accordance with the present invention. The host computer **102** is connected to an intelligent switch **142** . The switch **142** is connected to the primary data volume **104** and the data protection system **106** , which in turn manages the secondary data volume **108** . The switch **142** includes the ability to host applications and contains some of the functionality of the data protection system **106** in hardware, to assist in reducing system latency and improve bandwidth.

It is noted that the data protection system **106** operates in the same manner, regardless of the particular construction of the protected computer system **100** , **120** , **140** . The major difference between these deployment options is the manner and place in which a copy of each write is obtained. To those skilled in the art it is evident that other embodiments, such as the cooperation between a switch platform and an external server, are also feasible.

Conceptual Overview

To facilitate further discussion, it is necessary to explain some fundamental concepts associated with a continuous data protection system constructed in accordance with the present invention. In practice, certain applications require continuous data protection with a block-by-block granularity, for example, to rewind individual transactions. However, the period in which such fine granularity is required is generally short (for example, two days), which is why the system can be configured to fade out data over time. The present invention discloses data structures and methods to manage this process automatically.

The present invention keeps a log of every write made to a primary volume (a "write log") by duplicating each write and directing the copy to a cost-effective secondary volume in a sequential fashion. The resulting write log on the secondary volume can then be played back one write at a time to recover the state of the primary volume at any previous point in time. Replaying the write log one write at a time is very time consuming, particularly if a large amount of write activity has occurred since the creation of the write log. In typical recovery scenarios, it is necessary to examine how the primary volume looked like at multiple points in time before deciding which point to recover to. For example, consider a system that was infected by a virus. In order to recover from the virus, it is necessary to examine the primary volume as it was at different points in time to find the latest recovery point where the system was not yet infected by the virus. Additional data structures are needed to efficiently compare multiple potential recovery points.

Delta Maps

Delta maps provide a mechanism to efficiently recover the primary volume as it was at a particular point in time without the need to replay the write log in its entirety, one write at a time. In particular, delta maps are data structures that keep track of data changes between two points in time. These data structures can then be used to selectively play back portions of the write log such that the resulting point-in-time image is the same as if the log were played back one write at a time, starting at the beginning of the log.

FIG. 2 shows a delta map **200** constructed in accordance with the present invention. While the format shown in FIG. 2 is preferred, any format containing similar information may be used. For each write to a primary volume, a duplicate write is made, in sequential order, to a secondary volume. To create a mapping between the two volumes, it is preferable to have an originating entry and a terminating entry for each write. The originating entry includes information regarding the origination of a write, while the terminating entry includes information regarding the termination of the write.

As shown in delta map **200** , row **210** is an originating entry and row **220** is a terminating entry. Row **210** includes a field **212** for specifying the region of a primary volume where the first block was written, a field **214** for specifying the block offset in the region of the primary volume where the write begins, a field **216** for specifying where on the secondary volume the duplicate write (i.e., the copy of the primary volume write) begins, and a field **218** for specifying the physical device (the physical volume or disk identification) used to initiate the write. Row **220** includes a field **222** for specifying the region of the primary volume where the last block was written, a field **224** for specifying the block offset in the region of the primary volume where the write ends, a field **226** for specifying the where on the secondary volume the duplicate write ends, and a field **228** . While fields **226** and **228** are provided in a terminating entry such as row **220** , it is noted that field **226** is optional because this value can be calculated by subtracting the offsets of the originating entry and the terminating entry (field **226** =(field **224** −field **214**)+field **216**), and field **228** is not necessary since there is no physical device usage associated with termination of a write.

In a preferred embodiment, as explained above, each delta map contains a list of all blocks that were changed during the particular time period to which the delta map corresponds. That is, each delta map specifies a block region on the primary volume, the offset on the primary volume, and physical device information. It is noted, however, that other fields or a completely different mapping format may be used while still achieving the same functionality. For example, instead of dividing the primary volume into block regions, a bitmap could be kept, representing every block on the primary volume. Once the retention policy (which is set purely according to operator preference) no longer requires the restore granularity to include a certain time period, corresponding blocks are freed up, with the exception of any blocks that may still be necessary to restore to later recovery points. Once a particular delta map expires, its block list is returned to the appropriate block allocator for re-use.

Delta maps are initially created from the write log using a map engine, and can be created in real-time, after a certain number of writes, or according to a time interval. It is noted that these are examples of ways to trigger the creation of a delta map, and that one skilled in the art could devise various other triggers. Additional delta maps may also be created as a result of a merge process (called "merged delta maps") and may be created to optimize the access and restore process. The delta maps are stored on the secondary volume and contain a mapping of the primary address space to the secondary address space. The mapping is kept in sorted order based on the primary address space.

One significant benefit of merging delta maps is a reduction in the number of delta map entries that are required. For example, when there are two writes that are adjacent to each other on the primary volume, the terminating entry for the first write can be eliminated from the merged delta map, since its location is the same

as the originating entry for the second write. The delta maps and the structures created by merging maps reduces the amount of overhead required in maintaining the mapping between the primary and secondary volumes.

Data Recovery

Data is stored in a block format, and delta maps can be merged to reconstruct the full primary volume as it looked like at a particular point in time. Users need to be able to access this new volume seamlessly from their current servers. There are two ways to accomplish this at a block level. The first way is to mount the new volume (representing the primary volume at a previous point in time) to the server. The problem with this approach is that it can be a relatively complex configuration task, especially since the operation needs to be performed under time pressure and during a crisis situation, i.e., during a system outage. However, some systems now support dynamic addition and removal of volumes, so this may not be a concern in some situations.

The second way to access the recovered primary volume is to treat the recovered volume as a piece of removable media (e.g., a CD), that is inserted into a shared removable media drive. In order to properly recover data from the primary volume at a previous point in time, an image of the primary volume is loaded onto a location on the network, each location having a separate identification known as a logical unit number (LUN). This image of the primary volume can be built by using a method **300** to recover data by accessing a previously stored snapshot, as shown in FIG. 3.

The method **300** begins (step **302**) by selecting a snapshot for the primary volume to be recovered (step **304**). Since there will be multiple snapshots available for each protected volume, the actual snapshot which is required for access needs to be selected. A list of available snapshots for a particular protected volume can be displayed from a graphical user interface (GUI) of the data protection system. The snapshot to be selected can be either a scheduled snapshot or an any point in time (APIT) snapshot.

FIG. 4 shows a diagram of a retention policy used in connection with fading out the APIT snapshots over time. The retention policy consists of several parts. One part is used to decide how large the APIT window is and another part decides when to take scheduled snapshots and for how long to retain them. Each scheduled snapshot consists of all the changes up to that point in time; over longer periods of time, each scheduled snapshot will contain the changes covering a correspondingly larger period of time, with the granularity of more frequent snapshots being unnecessary.

The user can select any time that occurs within the APIT coverage. If the selected point in time occurs within a period in which the write log is out of sync (usually due to an earlier shutdown or error condition), then APIT snapshots for that period will not be available (the user will not be able to select a point in time for which the write log is out of sync). The list of times for which the write log is out of sync are determined by the data protection system and are saved with the primary volume.

Referring back to FIG. 3, a restore LUN is selected to load the snapshot onto (step **306**). The restore LUN is the method for accessing a snapshot from the host. In this role, the restore LUN acts as a virtual removable media disk device (e.g., a CD

drive) and the snapshot to be accessed acts as virtual piece of removable media (e.g., a CD). It is possible to restrict access to the restore LUN, permitting only authorized host computers to access the restore LUN. This type of access can be set via an access policy or other suitable access control mechanism.

Next, a determination is made whether the selected snapshot (from step **304**) is a scheduled snapshot or an APIT snapshot (step **308**). If the selected snapshot is a scheduled snapshot, then a point in time (PIT) map is created for the snapshot using a delta map manager (step **310**). Regions of all the delta maps prior to the time of the selected snapshot are merged to create the PIT map region by region. In order to enhance performance and the speed of access to a snapshot, the snapshot data can be accessed while the PIT map is being constructed. The snapshot is "loaded" onto the selected restore LUN (step **312**), and the method terminates (step **314**).

If the selected snapshot is an APIT snapshot (step **308**), then a new delta map is created covering the time between the time of the selected snapshot and the time of the delta map immediately preceding the time of the selected snapshot (step **316**). This new delta map is created because there is not necessarily a delta map corresponding to the time of the selected APIT snapshot. Due to the nature of APIT coverage, it is simply not feasible to store delta maps for every point in time in the APIT window. The procedure then continues with step **310** , as described above, with the new delta map being used in connection with the creation of the PIT map.

When an application or file system accesses a certain block on the restore LUN, the system uses the map to determine which block should be returned. If this particular block has not been resolved yet, the block is resolved immediately. Resolving a block refers to the map merging process. When a certain block has been "resolved," it means that through map merging it has been determined which block on the secondary volume should be presented to the host as part of the removable media. This creates the illusion to the user that the full volume has already been recreated. To avoid possible delays when accessing portions of the restore LUN, the user may request that the entire map be generated and loaded into memory. This will cause a longer delay before the first access, but creates a more predictable delay once the snapshot is mounted.

After the snapshot has been loaded onto the restore LUN, the user can access the snapshot as if it were the primary volume at the selected previous point in time. The snapshot is fully read/write accessible, and the user can perform a roll-forward of all the writes that occurred from the time of the snapshot. Changes made to the snapshot are not duplicated onto the primary volume, because the snapshot is, by definition, a reflection of the primary volume at a previous point in time. It is noted that while the user is accessing a snapshot, the primary volume is still being protected as under normal operating conditions. Furthermore, different snapshots can be loaded into different LUNs; the user is not restricted to accessing one snapshot at a time. Once the user is finished with the restore LUN(s), the GUI can be used to unload the snapshot or the snapshot can be ejected from the shared removable media drive by the host, similar to how a CD can be ejected.

Another important point to mention is that read/write access is important in this scenario. This is because when an application or even a journaled file system attempts to recover from the (possibly inconsistent) state the new volume presents, these applications need to be able to replay a log or perform other writes to the

volume. A system that does not offer read/write access is extremely limited in functionality. In the present invention, writes are stored in a temporary buffer, such that the original PIT image can be loaded again in its original state if desired.

In regard to performance optimization, it is not necessary to perform all of the delta map merges before the volume is presented to the host. Instead, the volume can be presented to the host immediately. Then, as the file system at the host accesses certain blocks, these can be resolved right away. The first time the system accesses a certain block may be slower because of this, but if the system accesses the same blocks again later, the access performance will have improved. While the host is not requesting new blocks, the system automatically continues to resolve the remaining maps. The map merging being performed in this instance relates to merging all the delta maps that are relevant to the selected PIT, to create a single map of all the blocks of the primary volume at that PIT.

Users should be able to browse files and folders and search for files with certain contents, even in the absence of a server. It is nonsensical to recover an entire 200 GB volume just to check if a specific file was already corrupted at a given point in time. The present invention is able to present volumes immediately, as discussed above. So the particular file can be examined and the remainder of the volume does not need to be resolved. But this still requires a server/file system.

The present invention also has the capability of decoding file system information and presenting the user with browsable list of files via FTP or a Web interface. This interface allows users to browse to a specific directory or file and then navigate to the previous/next (or any other) snapshot that was taken of the selected file. Only the necessary blocks will be resolved for this operation, and users are able to navigate through terabytes of data in a minimal amount of time to find the restore volume they are looking for or to just restore the file or directory they are trying to recover.

Automated searches can be performed in a similar fashion, such that the system could automatically find a certain file or content. For example, if a virus struck and corrupted the system, it is difficult to navigate many volumes by time. This is because the virus could have been there already for a long time. Executable files don't change over time, except when a virus strikes, so the system could be queried to find the point in time when the executable changed. Another useful query would be to see a list of different versions of the same file, including size and attributes. From the list, the user can immediately determine the time when the file was updated, for example, during an all-night work session, because it will include the greatest number of changes.

While specific embodiments of the present invention have been shown and described, many modifications and variations could be made by one skilled in the art without departing from the scope of the invention. The above description serves to illustrate and not limit the particular invention in any way.

<- Previous Patent (Method and apparatus...) | Next Patent (Data processing syst...) ->